



**UNIVERSIDAD CARLOS III DE MADRID  
ESCUELA POLITÉCNICA SUPERIOR**

**INGENIERÍA EN INFORMÁTICA  
PROYECTO FIN DE CARRERA**

***BIDAIRO, SERVIDOR PRIVADO DE RAGNAROK ONLINE***  
**(JUEGO DE ROL MULTIJUGADOR ONLINE MASIVO)**

**Autor:** Ricardo Iturria López

**Tutor:** Juan Peralta Donate

Octubre de 2010



*Dedicado a todos los miembros que han formado parte del Staff de BidaiRO y a los que todavía siguen en el barco, sin los cuales no hubiera sido posible llevar a cabo este proyecto.*

*A **Albion** y a su pobre corazón.*

*A **Badhip** y a su cabeza, donde quiera que esté.*

*A **Luffy**, al seno, y al coseno.*

*A **Pijus**, siempre deseosa de nuevos gorros.*

*A **Seiryu**, a sus funciones y a sus core dumps.*

*A **Zeus** y a su inigualable capacidad de destrucción.*

# 1    **Resumen**

En los últimos años, gracias a la evolución tecnológica y la gran mejora de las comunicaciones a través de Internet, los juegos online han obtenido cada vez más importancia en el mundo de los videojuegos. Existen muchos tipos de juegos online, algunos con raíces más antiguas que otros, entre los cuales se encuentran los juegos online multijugador, que han pasado de ser juegos relativamente sencillos en los que una pequeña cantidad de jugadores se reunía para jugar, a universos complejos donde pueden interactuar miles de personas al mismo tiempo. Hoy en día existe una gran variedad de estos juegos, y uno de ellos es el Ragnarok Online.

En esta memoria se hablará de lo que supone la creación y el mantenimiento de un servidor privado de Ragnarok Online desde un punto de vista tanto de jugador como de administrador. El servidor BidaiRO lleva en funcionamiento más de dos años y medio, por lo que se cuenta con una amplia experiencia en este campo.

## 2 Índice general

<b>1</b>	<b>RESUMEN</b>	<b>2</b>
<b>2</b>	<b>ÍNDICE GENERAL</b>	<b>3</b>
<b>3</b>	<b>ÍNDICE DE IMÁGENES</b>	<b>7</b>
<b>4</b>	<b>ÍNDICE DE TABLAS</b>	<b>9</b>
<b>5</b>	<b>INTRODUCCIÓN</b>	<b>10</b>
<b>5.1</b>	<b>POR QUÉ UN SERVIDOR DE RAGNAROK ONLINE</b>	<b>11</b>
<b>5.2</b>	<b>HISTORIA DE BIDAÍRO</b>	<b>12</b>
<b>5.3</b>	<b>OBJETIVOS</b>	<b>15</b>
<b>5.4</b>	<b>CONTENIDO DE LA MEMORIA</b>	<b>18</b>
<b>6</b>	<b>ESTADO DE LA CUESTIÓN</b>	<b>20</b>
<b>6.1</b>	<b>BREVE HISTORIA DE LOS VIDEOJUEGOS</b>	<b>21</b>
<b>6.2</b>	<b>LOS JUEGOS ONLINE MULTIJUGADOR</b>	<b>28</b>
6.2.1	POR E-MAIL	28
6.2.2	MULTI USER DUNGEONS (MUDs)	29
6.2.3	PRIMERA PERSONA	32
6.2.4	ESTRATEGIA EN TIEMPO REAL	33
6.2.5	POR NAVEGADOR	34
6.2.6	MASSIVE MULTIPLAYER ONLINE ROLE-PLAYING GAMES (MMORPGs)	35
<b>6.3</b>	<b>RAGNAROK ONLINE</b>	<b>38</b>
6.3.1	MECÁNICA DEL JUEGO	38
6.3.2	ATRIBUTOS DEL PERSONAJE	39
6.3.3	CLASES	42
6.3.4	CLANES	48
6.3.5	GUERRA DEL EMPERIUM	49
6.3.6	FUTURO DEL RAGNAROK ONLINE	51

<b>6.4</b>	<b>SERVIDORES DE RAGNAROK ONLINE</b>	<b>54</b>
6.4.1	OFICIALES	54
6.4.2	PRIVADOS	55
<b>7</b>	<b>ESTRUCTURA DEL SERVIDOR BIDAIRO</b>	<b>58</b>
<b>7.1</b>	<b>PORTAL WEB</b>	<b>59</b>
7.1.1	MAPA WEB	62
7.1.2	SECCIÓN <i>PRINCIPAL</i>	64
7.1.3	SECCIÓN <i>INFO SERVER</i>	65
7.1.4	SECCIÓN <i>DESCARGAS</i>	66
7.1.5	SECCIÓN <i>CÓMO JUGAR</i>	67
7.1.6	SECCIÓN DE CALCULADORAS	68
7.1.7	PANEL DE CONTROL	69
7.1.8	FORO	70
7.1.9	MAPA DE INTERACCIÓN	72
<b>7.2</b>	<b>SERVIDOR eATHENA</b>	<b>75</b>
7.2.1	LOGIN SERVER	77
7.2.2	CHAR SERVER	79
7.2.3	MAP SERVER	79
7.2.4	ANTICHEAT SERVER	81
<b>7.3</b>	<b>CLIENTE</b>	<b>84</b>
7.3.1	CLIENTE DE GRAVITY	84
7.3.2	CLIENTE BIDAIRO	84
7.3.3	GRFs	86
7.3.4	PARCHEADOR	90
<b>7.4</b>	<b>SERVIDOR MYSQL</b>	<b>94</b>
7.4.1	BASES DE DATOS Y TABLAS	94
7.4.2	USUARIOS Y PERMISOS	101
<b>8</b>	<b>PAUTAS PARA LA GESTIÓN</b>	<b>106</b>
<b>8.1</b>	<b>PAUTAS PARA LA GESTIÓN</b>	<b>107</b>
8.1.1	GESTIÓN DE CAPACIDAD	107
8.1.2	ESCALABILIDAD	108

8.1.3	GESTIÓN DE LA CONTINUIDAD	109
8.1.4	TIEMPO DE INCIDENTE	109
8.1.5	DISPONIBILIDAD Y TIEMPO DE RESPUESTA	112
<b>9</b>	<b><u>CONCLUSIONES</u></b>	<b><u>113</u></b>
9.1	CONCLUSIONES	114
<b>10</b>	<b><u>LÍNEAS FUTURAS</u></b>	<b><u>116</u></b>
10.1	LÍNEAS FUTURAS	117
10.1.1	SERVIDOR PK	117
10.1.2	RAZAS	120
10.1.3	SERVIDORES MÚLTIPLES	122
10.1.4	PORTAL INFORMATIVO	123
<b>11</b>	<b><u>PLANIFICACIÓN Y COSTES</u></b>	<b><u>126</u></b>
11.1	PLANIFICACIÓN Y COSTES	127
11.1.1	FASE 1: CREACIÓN DEL PROYECTO	127
11.1.2	FASE 2: MANTENIMIENTO DEL PROYECTO	139
11.2	PLAN DE NEGOCIO	141
<b>12</b>	<b><u>ANEXO A: TRADUCCIONES DE LAS CLASES DEL RAGNAROK</u></b>	<b><u>144</u></b>
<b>13</b>	<b><u>ANEXO B: ESTRUCTURA DEL FORO</u></b>	<b><u>146</u></b>
<b>14</b>	<b><u>ANEXO C: SCRIPT DE ACTUALIZACIÓN DE EATHENA</u></b>	<b><u>149</u></b>
<b>15</b>	<b><u>ANEXO D: IMPLEMENTACIÓN DEL TRIVIAL ONLINE</u></b>	<b><u>151</u></b>
<b>16</b>	<b><u>ANEXO E: IMPLEMENTACIÓN DE EVENTOS</u></b>	<b><u>157</u></b>
<b>17</b>	<b><u>ANEXO F: PASOS PARA LA CREACIÓN DE UN CLIENTE</u></b>	<b><u>159</u></b>
<b>18</b>	<b><u>ANEXO G: IMPLEMENTACIÓN DEL PARCHEADOR</u></b>	<b><u>160</u></b>

<b><u>19</u></b>	<b><u>ANEXO H: FICHEROS UTILIZADOS POR EL PARCHEADOR</u></b>	<b><u>166</u></b>
<b><u>20</u></b>	<b><u>ANEXO I: CONFIGURACIÓN DEL SERVIDOR MYSQL</u></b>	<b><u>168</u></b>
<b><u>21</u></b>	<b><u>ANEXO J: ANÁLISIS DE RENDIMIENTO DE MYSQL</u></b>	<b><u>174</u></b>
<b><u>22</u></b>	<b><u>DICCIONARIO DE TÉRMINOS Y ACRÓNIMOS</u></b>	<b><u>178</u></b>
<b><u>23</u></b>	<b><u>REFERENCIAS</u></b>	<b><u>191</u></b>



### 3 Índice de imágenes

IMAGEN 1: VIDEOJUEGO DE LCD.....	24
IMAGEN 2: CAPTURA DE PANTALLA DE UN MUD .....	30
IMAGEN 3: ÁRBOL DE FAMILIA DE DIKUMUD .....	31
IMAGEN 4: CONTINENTE RUNE MIDGARD.....	39
IMAGEN 5: VENTANA DE STATS Y EQUIPO.....	42
IMAGEN 6: TABLA CON LOS DISTINTOS JOBS.....	47
IMAGEN 7: VENTANA DE GUILD Y PERSONAJE CON EL EMBLEMA JUNTO A SU NOMBRE .....	49
IMAGEN 8: UN CASTILLO CON UN EMPERIUM.....	51
IMAGEN 9: RAGNAROK ONLINE 2: THE GATE OF THE WORLD.....	53
IMAGEN 10: EL TOP DE DEVSTORM NETWORK.....	56
IMAGEN 11: PÁGINA PRINCIPAL DE BIDAIRÓ .....	61
IMAGEN 12: BIDAIRÓ EN LA PRIMERA POSICIÓN DEL GOOGLE .....	62
IMAGEN 13: MAPA DE LA WEB DE BIDAIRÓ.....	63
IMAGEN 14: BUGTRACKER .....	70
IMAGEN 15: REPORTE DE UN BUG EN EL BUGTRACKER .....	71
IMAGEN 16: ESTRUCTURA DEL PORTAL WEB.....	73
IMAGEN 17: DIAGRAMA DE SECUENCIA DE CREACIÓN DE CUENTA.....	74
IMAGEN 18: INTERACCIÓN BÁSICA DEL SERVIDOR EATHENA EN BIDAIRÓ.....	77
IMAGEN 19: ESTRUCTURA BÁSICA DE UN LOGIN SERVER CON MÚLTIPLES CHAR/MAP SERVERS .....	78
IMAGEN 20: MAP SERVER B COMUNICÁNDOSE CON EL RESTO DE MAP SERVERS .....	80
IMAGEN 21: CONTENIDO DE LA TABLA ANTICHEAT.....	81
IMAGEN 22: FLUJO DETALLADO DE LA COMUNICACIÓN ENTRE EL ANTICHEAT Y EATHENA.....	82
IMAGEN 23: DIAGRAMA DE SECUENCIA DE LOGIN.....	83
IMAGEN 24: VENTANA PRINCIPAL DE DIFFPATCHER .....	85
IMAGEN 25: BIDAI.GRF VISTO DESDE GRF TOOL.....	89
IMAGEN 26: PARCHE EDITADO CON GRYFF .....	89
IMAGEN 27: APLICACIÓN PRIVADA PARA PARCHEAR.....	90
IMAGEN 28: APLICACIÓN PRIVADA PARA CALCULAR CRCs .....	90
IMAGEN 29: PARCHEADOR CON LA VENTANA DE PERSONALIZACIÓN .....	93
IMAGEN 30: EJEMPLO DE ELFO OSCURO .....	121
IMAGEN 31: EJEMPLO DE NO-MUERTO .....	122
IMAGEN 32: EJEMPLO DE ORCO.....	122
IMAGEN 33: BASE DE DATOS DE RATEMYSERVER.NET .....	124
IMAGEN 34: PLANIFICACIÓN INICIAL FASE 1 (1 DE 2) .....	132
IMAGEN 35: PLANIFICACIÓN INICIAL FASE 1 (2 DE 2) .....	133
IMAGEN 36: PLANIFICACIÓN FINAL FASE 1 (1 DE 2).....	136

IMAGEN 37: PLANIFICACIÓN FINAL FASE 1 (2 DE 2).....137

IMAGEN 38: DIAGRAMA DE GANTT FASE 2.....140

IMAGEN 39: PARTIDA DE TRIVIAL ONLINE .....152

IMAGEN 40: EVENTO DEL VAMPIRO MATÍAS.....158

IMAGEN 41: DIAGRAMA PARA LA CREACIÓN DE UN CLIENTE .....159

IMAGEN 42: MAPA DE MÉTODOS DEL PARCHEADOR (PRINCIPAL).....163

IMAGEN 43: MAPA DE MÉTODOS DEL PARCHEADOR (PERSONALIZACIÓN) .....165

## 4 Índice de tablas

TABLA 1: SERVIDORES OFICIALES DE RO .....	54
TABLA 2: PRECIOS DE LOS SERVIDORES OFICIALES IRO Y EURO.....	55
TABLA 3: SUBDOMINIOS WEB DE BIDAIRIO .....	59
TABLA 4: ORDEN DE CARGA DE LOS GRF.....	87
TABLA 5: FICHEROS ACTUALIZABLES A TRAVÉS DEL PARCHEADOR .....	91
TABLA 6: ACCIONES REALIZADAS POR EL PARCEHADOR DEPENDIENDO DEL FICHERO.....	92
TABLA 7: LISTADO DE BASES DE DATOS UTILIZADAS .....	95
TABLA 8: TABLAS DE BIDAIRIO PARA LA BASE DE DATOS RAGNAROK .....	99
TABLA 9: TABLAS DE BIDAIRIO PARA LA BASE DE DATOS RAGNAROK_LOG .....	100
TABLA 10: TABLAS DE BIDAIRIO PARA LA BASE DE DATOS RAGNAROK_FORO.....	100
TABLA 11: TABLAS DE BIDAIRIO PARA LA BASE DE DATOS TS.....	100
TABLA 12: USUARIOS Y PERMISOS DE LAS BASES DE DATOS.....	105
TABLA 13: CONSUMO DE RECURSOS DEL SERVIDOR .....	107
TABLA 14: LISTA DE RAZAS .....	121
TABLA 15: CARACTERÍSTICAS DEL HOST (MÁQUINA Y RED) .....	130
TABLA 16: COSTE DEL SERVIDOR FASE 1 (1 DE 2).....	138
TABLA 17: COSTE DEL SERVIDOR FASE 1 (2 DE 2).....	138
TABLA 18: COSTE DEL SERVIDOR FASE 2 (1 DE 2).....	140
TABLA 19: COSTE DEL SERVIDOR FASE 2 (2 DE 2).....	140
TABLA 20: BENEFICIOS MENSUALES.....	142
TABLA 21: BENEFICIOS DIARIOS .....	142
TABLA 22: TRADUCCIÓN INGLÉS-ESPAÑOL DE LAS CLASES DEL RAGNAROK .....	145
TABLA 23: DIVISIÓN EN FICHEROS DEL CÓDIGO DEL TRIVIAL .....	153

## **5    Introducción**

## 5.1 *Por qué un servidor de Ragnarok Online*

Desde hace algunos años, con el auge de Internet, los juegos online han proliferado y se han convertido en un elemento más en la vida diaria de muchas personas. Gracias a la posibilidad de aplicar una gran variedad de tecnología de forma sencilla y a la facilidad de acceso a los contenidos online de la que se dispone hoy en día, muchas empresas se han embarcado en la creación de mundos virtuales donde la gente puede interactuar y colaborar para desempeñar distintos objetivos tanto individuales como colectivos.

Una de las empresas dedicadas a esto es Gravity<sup>[1]</sup>, de origen coreano, que el 31 de agosto de 2002 lanzó en Corea el **Ragnarok Online**<sup>[2]</sup> (RO), su juego de rol multijugador online masivo (MMORPG<sup>[1][3]</sup>) ambientado principalmente en la mitología nórdica (el Ragnarok es el fin del mundo, que a su vez permitirá el resurgir de uno nuevo<sup>[4]</sup>), pero con influencias de otras muchas culturas. Aunque no fue hasta abril de 2004 cuando el juego hizo su aparición a nivel europeo y ganó más popularidad entre los jugadores occidentales<sup>[5]</sup>.

Un MMORPG se compone de uno o más servidores oficiales, creados y mantenidos por la compañía propietaria del juego (Gravity en el caso del RO), a los cuales los jugadores se conectan para poder jugar. Al igual que con otros juegos de este tipo, aparte de los oficiales existen multitud de servidores privados creados por aficionados, en los que no es necesario pagar una cuota mensual para jugar (aunque es muy común aceptar donaciones y otorgar ciertos beneficios a la gente que decide pagar), y suelen tener características distintas a los oficiales ya que las reglas las pone quien los crea.

BidaiRO es uno de estos servidores privados, abierto desde finales de febrero de 2008, muy conocido en España, México y Sudamérica debido a su calidad y dedicación por parte de la administración. El motivo de su apertura fue tener un buen servidor de rates<sup>[2]</sup> bajos donde poder jugar, ya que la gran mayoría de los servidores privados de estas características cierra a los pocos meses de abrir, están mal gestionados, hay corrupción y en general no ofrecen un servicio aceptable para los más exigentes.

## **5.2      *Historia de BidaiRO***

El servidor abrió sus puertas a finales de febrero de 2008, pero fue en septiembre de 2007 cuando se decidió crear un servidor propio y posteriormente a buscar un host<sup>[3]</sup> donde alojarlo. Desde entonces BidaiRO ha pasado por varias etapas y ha superado muchos baches hasta el día de hoy.

Una de las dificultades de un proyecto de estas características son los propios usuarios. Normalmente la media de edad puede estar entre los 13 – 22 años (aunque siempre hay excepciones), y esto hace que en muchos casos sea complicado tratar con ellos. Por poner un ejemplo, en los inicios del servidor un usuario fue descubierto haciendo trampas y al aplicársele la sanción se fue del servidor con sus amigos, repercutiendo en el número de usuarios y creando mala fama contando su visión de los hechos, a pesar de que el resto de usuarios estuvieran muy contentos.

Para evitar esto se implementó un sistema que evitara ese tipo de trampas y durante varios meses la vida del servidor fue buena, con un número de usuarios aceptable y muchas novedades e implementaciones exclusivas por parte de la dirección. Desgraciadamente en noviembre de 2008 la empresa de hosting se fue a la quiebra y dejó de prestar servicio de un día para otro sin previo. Fue necesario buscar otro host y volver a comenzar de nuevo utilizando copias de seguridad, pero por desgracia los datos de los jugadores se perdieron, puesto que sólo existían copias en la propia máquina del servidor y ya no se tenía acceso a ella.

En ese momento se decidió continuar con el proyecto, aunque dos de sus integrantes lo abandonaron, por lo que se redujo la plantilla. Aun así a finales de diciembre de 2008 se comenzó una nueva etapa de BidaiRO con el apoyo de muchos de los antiguos usuarios. A partir de entonces las copias de seguridad comenzaron a guardarse tanto en el propio host como en una máquina externa para evitar su pérdida en caso de repetirse otro problema grave con el host.

Que un servidor empiece una nueva etapa no suele estar bien visto porque supone la pérdida de todos los personajes de los jugadores, sin embargo resultó algo

positivo. En este tipo de juegos cuando un jugador alcanza el máximo nivel y no le queda mucho por hacer, pueden pasar dos cosas: que se aburra y busque otro juego o que encuentre la forma de divertirse con lo que tiene y las opciones que le ofrece el Ragnarok Online (por ejemplo crear un nuevo personaje y utilizar el viejo para conseguir objetivos más difíciles). Normalmente los jugadores de servidores privados no se llegan a aburrir porque el servidor cierra antes de que esto ocurra, pero en este caso antes del incidente del host se notó un descenso en el número de usuarios que tenían personajes de máximo nivel. Empezar una nueva etapa, obligando a todos a comenzar de cero, hizo que los usuarios volvieran a tener interés por el juego, y también atrajo a otros que preferían empezar en un servidor con jugadores a su mismo nivel (algo también muy común),

Aprovechando la segunda etapa del servidor, también se cambiaron algunas de sus características que lo diferenciaban de los servidores oficiales que parecían echar atrás a nuevos jugadores. Esto hizo que la similitud con el servidor oficial fuera mayor y vinieran nuevos usuarios que en la etapa anterior no llegaron a decidirse.

Durante varios meses el servidor fue muy bien, con una gran cantidad de usuarios, pero había otro problema. Todo servidor privado de Ragnarok Online utiliza un emulador<sup>[4]</sup> de código abierto<sup>[6]</sup> que ofrece las características de los servidores oficiales de la forma más fiel posible. El emulador utilizado por BidaiRO se obtiene mediante un repositorio SVN<sup>[5][7]</sup>, en el cual hay dos ramas principales: *trunk* y *stable*. La primera equivale a un entorno de desarrollo<sup>[6]</sup>, y la segunda a un entorno de producción<sup>[7]</sup>.

En BidaiRO se utilizaba la rama *stable* para ofrecer a los usuarios un entorno libre de errores, pero los desarrolladores del emulador decidieron dejar de actualizar esta rama y trabajar de forma casi exclusiva en la *trunk*, informando vagamente de esta decisión unos meses después de haberla tomado. Esto supuso que el servidor quedara desactualizado durante casi medio año, lo que hizo que casi la totalidad de los usuarios migraran a otros servidores mucho más actualizados que utilizaban la rama *trunk* antes de poder reaccionar.

La única forma de conseguir que el servidor volviera a crecer fue cerrarlo temporalmente para fusionar la rama stable con la trunk, que por aquel entonces ya era mucho más estable que antes, y abrir comenzando una tercera etapa, que es la que se está jugando actualmente. Por aquel entonces el proyecto sólo contaba con tres miembros, uno de ellos sin conocimientos técnicos y otro que apenas estaba activo. Aun así se decidió continuar y se pudieron fusionar las dos ramas en poco tiempo, abriendo la tercera etapa a finales de junio de 2009.

En esta tercera etapa se volvieron a cambiar otras características que lo diferenciaban ligeramente de los servidores oficiales, siendo todavía más fiel que antes, y se cambió la política de actualizaciones de forma que se empezó a utilizar la rama trunk en vez de la stable, actualizando el emulador como mínimo una vez al mes.

Posiblemente esta sea la etapa en la que más usuarios ha tenido el servidor, aunque a día de hoy se haya reducido el número aproximadamente a la mitad debido al tiempo que lleva abierto, algo que se sabía que iba a ocurrir porque forma parte del ciclo de vida de un servidor de estas características. Pese a ello el servidor sigue contando con un número de usuarios suficiente para mantenerlo abierto, que disfrutan de implementaciones novedosas de forma periódica y muchos de los cuales se ofrecen a colaborar con el servidor mediante donaciones, ideas para mejorarlo y atraer nuevos usuarios, publicidad en foros y trabajos de diseño gráfico para la página web y el propio juego.

Todo lo que empieza tiene que tener un final, y posiblemente no se vaya a comenzar con una cuarta etapa de BidaiRO debido al esfuerzo que supone gestionarlo. Se seguirá manteniendo esta con toda la dedicación de siempre hasta que el número de usuarios caiga, y entonces se dará por terminado uno de los pocos proyectos españoles de estas características que ha logrado ofrecer un servicio magnífico durante más de dos años y medio.



### 5.3 **Objetivos**

Dado el tiempo que lleva abierto el servidor, se han ido añadiendo objetivos una vez se cumplían los anteriores. A continuación se muestran los principales:

- **Mantener el servidor abierto durante 1 año:** Dado que la mayoría de los servidores españoles con las características de BidaiRO suelen cerrar a los pocos meses de abrir, un objetivo importante fue el de conseguir que durara al menos un año. La idea es tener un servidor estable donde los usuarios no tuvieran que preocuparse porque fuera a cerrar cuando menos se lo esperasen. Con el servidor funcionando desde febrero de 2008, este objetivo se ha cumplido con creces.
- **Tener una mecánica fiel a la de los servidores oficiales:** Es muy común que los servidores privados tengan características personalizadas que los hagan diferentes a los oficiales. Muchas de estas características se consideran “trampas” al compararlas con los oficiales, por ejemplo la posibilidad de reestructurar las características del personaje en cualquier momento después de haberlo creado, y otras son personalizaciones de otros tipos como pueden ser ítems exclusivos de ese servidor. En BidaiRO no hay ese tipo de cosas, por lo que la experiencia de juego es lo más parecida posible a estar en un servidor oficial.
- **Implementaciones exclusivas:** Esto viene ligado al punto anterior. En BidaiRO se quieren mantener las características de los servidores oficiales, pero también se quieren implementar cosas que hagan al servidor único. Para ello se han buscado opciones alternativas al juego normal que no tengan repercusión directa en este, de forma que los usuarios puedan divertirse con cosas que no encontrarán en otros servidores sin que influya en la forma de juego pensada por Gravity, como puede ser un trivial online, conquistas de ciudades o combates amistosos entre jugadores utilizando personajes de máximo nivel creados sólo para ese fin.

- **No dar ventajas desequilibrantes a los donantes:** Aunque todos los servidores privados son gratuitos, en muchos donar implica obtener ciertas ventajas sobre los que no donan. Esto es algo que nunca se ha querido en BidaiRO, por lo que las ventajas por donar se traducen en la posibilidad de obtener objetos decorativos como gorros, colores de ropa especiales y cosas similares que no afectan al juego normal. Muchos usuarios donan para poder cambiar la apariencia de sus personajes y otros para ayudar al servidor, pero nunca nadie se sentirá obligado a donar para poder equipararse a otra persona que haya pagado para conseguir un personaje más poderoso.
- **Dar un buen soporte al usuario:** Algo de lo que carecen muchos servidores es de un buen soporte y trato con el usuario. En este servidor siempre se ha perseguido contestar a las dudas, resolver problemas y gestionar cualquier cosa que el usuario pudiera querer de forma rápida y eficaz, y esto es algo en lo que todos los usuarios de BidaiRO están de acuerdo. Para ello se han habilitado foros y formularios donde el usuario puede expresar cualquier problema e inquietud y los administradores poder responder a la mayor brevedad posible.
- **Automatización de eventos<sup>[8]</sup>:** Los eventos siempre fueron un punto flaco del servidor. Generalmente a los usuarios les gusta que haya administradores presentes en el juego y organicen eventos, que suelen consistir en invasiones a ciudades, juegos del escondite donde debe buscarse al administrador, etc. En BidaiRO se han realizado pocos eventos de este tipo, ya que para organizarlos y realizarlos se requiere un tiempo del que generalmente no se dispone. Para ello, poco a poco se han implementado eventos diarios y automáticos que no requieren de la presencia de un administrador, hasta disponer de eventos fijos todos los días de la semana y alguno en días aleatorios.
- **Razas:** En el Ragnarok Online todos los personajes son humanos. Implementar una serie de razas típicas de este tipo de juegos, como pueden ser elfos, orcos, gigantes, etc. le daría variedad y sería algo que destacaría entre otros servidores (existen razas en algunos de ellos, pero no es lo normal). La implementación de

las razas sería puramente visual, para no interferir en la mecánica normal del juego.

- **Portal informativo:** En un juego tan extenso como el Ragnarok, es indispensable disponer de algún tipo de referencia a la que poder dirigirse para consultar toda la información posible, como pueden ser características de objetos, armas y armaduras, tipos de monstruos y localización de los mismos, tipos de tiendas y dónde encontrarlas, etc. Esta información se encuentra disponible en algunas páginas web creadas a tal efecto y que son visitadas asiduamente por muchos jugadores, pero disponer de una propia para el servidor es útil para hacerlo más conocido e incluso para hacer crecer su comunidad de usuarios.

## **5.4      *Contenido de la memoria***

A continuación se muestra una breve descripción de los capítulos que componen este documento.

### **Capítulo 1: Introducción**

En este capítulo se tratan las motivaciones para realizar un servidor de Ragnarok Online, la historia del servidor BidaiRO en los más de dos años y medio que lleva funcionando, y los objetivos que se plantearon al comenzar con el proyecto.

### **Capítulo 2: Estado de la cuestión**

En este capítulo se realiza un recorrido a través de la historia de los videojuegos, desde sus inicios a finales de la década de los 40 hasta hoy. Se podrá ver cómo los videojuegos online han surgido y evolucionado a partir de expuesto en el punto anterior y se realizará un breve análisis de los tipos principales de videojuegos online existentes, tanto actuales como otros más antiguos. Por último se presentarán los aspectos básicos que componen el Ragnarok Online en cuanto a modo de juego y características de este, así como una descripción de los aspectos y diferencias más características de los servidores públicos y privados.

### **Capítulo 3: Estructura del servidor BidaiRO**

En este capítulo se mostrarán aspectos más técnicos y concretos del servidor BidaiRO, divididos en cuatro grandes bloques: el portal web, el emulador (la parte del juego que se ejecuta en el servidor), el cliente (la parte del juego que se ejecuta en el ordenador del usuario) y el motor de bases de datos, el cual sirve de puente entre casi todos los componentes del servidor.

### **Capítulo 4: Conclusiones**

Aquí se analizará lo que ha supuesto llevar a cabo un proyecto de varios años y las posibilidades que se pueden abrir gracias a la experiencia adquirida en varios campos.

## **Capítulo 5: Líneas futuras**

Un proyecto de este tipo puede, y es incluso de esperar, seguir creciendo. En este capítulo se comentarán varias ideas que pueden implementarse progresivamente para mejorar el servidor, así como formas de expansión para llevar a cabo otros proyectos paralelos similares a este

## **Capítulo 6: Planificación y costes**

En este capítulo se verá cómo se ha planificado el proyecto dividiéndolo por fases, una de creación y otra de mantenimiento, y el coste que ha supuesto tanto inicialmente como mensualmente.

## **Anexos**

Se disponen de varios anexos donde podrán profundizarse algunos aspectos concretos del proyecto analizados previamente.

## **Diccionario de términos y acrónimos y Referencias**

Finalmente se podrán consultar todos los términos y acrónimos de los que se hace uso a lo largo del documento, así como las referencias utilizadas para documentarse en varios aspectos del mismo y otros artículos extra por si el lector quiere profundizar en algo.

## **6 Estado de la cuestión**

## 6.1 *Breve historia de los videojuegos*

El origen de los videojuegos se remonta al 25 de enero de 1947, cuando Thomas T. Goldsmith, Jr. y Estle Ray Mann patentaron un sistema que, mediante válvulas y un tubo de rayos catódicos<sup>[9]</sup>, simulaba el lanzamiento de un misil a un objetivo. Aunque esto no fue un videojuego propiamente dicho, se considera el primer paso para su desarrollo. Más cercano a un videojuego estaría el **OXO**<sup>[10]</sup> de A. S. Douglas, una implementación del tres en raya creada en 1952 para su tesis en la Universidad de Cambridge, en donde un jugador podía competir contra el ordenador.

En la **década de los 50 y de los 60** comenzaron a desarrollarse videojuegos más elaborados para aquel entonces, inicialmente programados como hobby por universitarios y no muy exitosos, ya que no existían ordenadores en los hogares y las universidades eran los únicos lugares donde se podía contar con equipos adecuados. Fue en 1961 cuando nació **Spacewar!**<sup>[11]</sup> de la mano de un grupo de estudiantes del MIT<sup>[9]</sup>, en el cual dos jugadores se enfrentaban a bordo de naves espaciales capaces de disparar misiles. La peculiaridad de este juego reside en que fue el primero en distribuirse con el ordenador para el que fue desarrollado, el **DEC PDP-1**<sup>[12]</sup>.

En 1966, Ralph Baer y Bill Harrison crearon **Chase**, un videojuego sencillo pero capaz de visualizarse en un televisor. Tras esto crearon la pistola de luz<sup>[13]</sup>, como la que se usa en el conocido juego de Nintendo<sup>[14][15]</sup> **Duck Hunt**<sup>[16]</sup>, consistente en cazar patos disparando a la pantalla con la pistola, y desarrollaron varios videojuegos para ella. Así, en 1969, nació la primera consola<sup>[17]</sup>.

En la **década de los 70** empezaron a proliferar los **arcades**<sup>[10]</sup> o máquinas recreativas, las **consolas** de primera y segunda generación y los **ordenadores personales**. En septiembre de 1971 se creó **Galaxy Game**<sup>[18]</sup>, el primer arcade basado en Spacewar!, pero sólo se produjo una unidad. Posteriormente, en el mismo año, Nolan Bushnell y Ted Dabney crearon otra versión arcade de Spacewar! a partir de la cual se crearon 1.500 máquinas más. Aunque no tuvo mucho éxito debido a la complejidad del juego, fue la primera vez que un arcade se produjo en masa. Al ver que su arcade no les reportaba beneficios, Bushnell y Dabney fundaron **Atari**<sup>[19]</sup> y en 1972 lanzaron el

arcade de **Pong**<sup>[20]</sup>, con el cual tuvieron mucho éxito y vendieron 19.000 máquinas. Fue a partir de entonces cuando, en 1978, los arcades comenzaron su época dorada con títulos como **Space Invaders**<sup>[21]</sup>, **Asteroids**<sup>[22]</sup>, **Pac-Man**<sup>[23]</sup>...

La primera consola se comenzó a vender en 1972 bajo el nombre de **Magnavox Odyssey**<sup>[24]</sup>, de la compañía Magnavox<sup>[25]</sup>. Los juegos estaban almacenados en la propia consola y se podía escoger a cuál jugar mediante cartuchos que modificaban unos jumpers<sup>[11]</sup>. Esto quiere decir que la consola estaba limitada a los juegos que ofrecía al comprarla. También tenía varias pegatinas para colocarlas en el monitor de forma que se pudieran simular escenarios para los juegos. Esta consola vendió 100.000 unidades sólo en el primer año, alcanzando un total de dos millones a lo largo de toda su producción.

A partir de 1977 llegaron las consolas de segunda generación, las cuales dejaron de venir con juegos de serie y en su lugar se programaron en los propios cartuchos, de forma que desaparecía la limitación existente en las consolas de primera generación. En esta generación fueron famosas la **Atari 2600**<sup>[26]</sup> de Atari, **Intellivision**<sup>[27]</sup> de Mattel<sup>[28][29]</sup> y **ColecoVision**<sup>[30]</sup> de Coleco<sup>[31][32]</sup>.

En cuanto a los ordenadores personales, hasta la década de los 80 no tuvieron demasiada relevancia para los videojuegos. La mayoría se desarrollaban en los equipos informáticos de las universidades, pero eran poco conocidos. Era normal distribuir el código fuente impreso en libros o revistas, aunque también había programadores, que vendían de vez en cuando sus discos por correo, aunque los programaban como afición y no se llegaban a comercializar.

Sin embargo, **en los años 80** surgen los primeros ordenadores personales diseñados principalmente para los videojuegos. Los arcades continúan siendo muy populares, pero las ventas de las consolas descienden debido a estos nuevos ordenadores, que cuentan con las mismas características que las consolas y son prácticamente igual de sencillos de utilizar debido a que están diseñados para los videojuegos. Los ordenadores más conocidos de esa época fueron el **Commodore 64**<sup>[33][34]</sup> de Commodore International<sup>[35][36]</sup>, **Apple II**<sup>[37]</sup> de Apple<sup>[38][39]</sup>, **ZX Spectrum**<sup>[40]</sup> de Sinclair Research Ltd.<sup>[41][42]</sup> y **Amstrad CPC**<sup>[43]</sup> de Amstrad<sup>[44][45][46]</sup>.



Ya entonces se comercializaban los primeros **PCs** compatibles<sup>[12]</sup>, pero debido a su baja calidad de vídeo y sonido y a su alto precio, no fueron tan populares como los anteriores. Esto empezó a cambiar a partir de 1985 cuando, a pesar de haber salido al mercado los **ST**<sup>[47]</sup> de Atari y los **Amiga**<sup>[48]</sup> de Commodore con arquitecturas de 16 bits (frente a las de 8 que había por aquel entonces), los PCs comenzaron a mejorar rápidamente sus deficiencias de audio y vídeo gracias a la fabricación de tarjetas de vídeo y de audio diseñadas para estos equipos. Esto, acompañado de una importante bajada de precios, hizo que los PCs comenzaran a imponerse incluso frente a estas nuevas máquinas de 16 bits: pese a tener unos juegos de gran calidad, su precio era demasiado elevado.

A mediados de esta década surgen las consolas de tercera generación, con la **Nintendo Entertainment System**<sup>[49]</sup> (*NES*) de Nintendo a la cabeza de los mercados norteamericanos y japoneses, o la **Master System**<sup>[50]</sup> de Sega<sup>[51]</sup> en Europa, Australia y Brasil. Esta nueva generación de consolas fue la primera en incorporar el **pad**<sup>[13]</sup> en sustitución de los tradicionales joysticks u otros dispositivos más rudimentarios.

También de la mano de Nintendo, antes incluso del lanzamiento de la NES, surgieron los primeros videojuegos de **LCD**<sup>[14]</sup>, en muchas ocasiones réplicas de otros juegos conocidos y que no llegaban a ser consolas portátiles debido a su extrema sencillez, pero que debido a su tamaño y funcionamiento con pilas podían llevarse a cualquier parte. En la *Imagen 1: Videojuego de LCD* se puede observar un ejemplo.



Imagen 1: Videojuego de LCD

En la **década de los 90** la calidad de los videojuegos comenzó a dar un salto gracias a las bajadas de precio de los procesadores, tarjetas gráficas y de sonido y al aumento de potencia que ofrecían. Los juegos ya no se limitaban al equipo de programadores como antes, sino que también empezó a participar la industria de la música y de la animación. De los antiguos videojuegos en 2D<sup>[15]</sup> se empezaron a hacer varios en 3D<sup>[16]</sup>, la perspectiva cambió a primera persona y se empezaron a popularizar los primeros juegos online multijugador. El concepto de **shareware**<sup>[17]</sup> se hizo más popular y muchos pequeños desarrolladores lo adoptaron para publicar sus juegos. Además se crearon sagas muy conocidas con videojuegos como **Alone in the Dark**<sup>[52]</sup>,

**King's Quest**<sup>[53]</sup>, **Monkey Island**<sup>[54]</sup>, **Myst**<sup>[55]</sup>, **SimCity**<sup>[56]</sup> y demás simuladores de Maxis<sup>[57]</sup>, **Heroes of Might and Magic**<sup>[58]</sup>, etc.

Así como los ordenadores personales, las consolas también dieron un gran salto. En 1989 Nintendo pone a la venta la **Game Boy**<sup>[59]</sup>, la primera consola portátil del mercado. La siguieron la Game Gear<sup>[60]</sup> de Sega y la Lynx<sup>[61]</sup> de Atari, de mejores prestaciones pero menos exitosas, sobre todo esta última, posiblemente por su tamaño y por la duración de la batería. Incluso en el campo de la telefonía móvil Nokia<sup>[62][63]</sup> innovó en 1998 con la inclusión del popular Snake<sup>[64]</sup> en sus teléfonos móviles.

La cuarta generación de consolas llegó en 1989 con la **Mega Drive**<sup>[65]</sup> de Sega, y fue seguida de la **Super NES**<sup>[66]</sup> de Nintendo, la **TurboGrafx-16**<sup>[67]</sup> de NEC<sup>[68][69]</sup> y la **Neo-Geo**<sup>[70]</sup> de SNK<sup>[71][72]</sup>. Estas consolas contaban con procesadores de 16 bits, lo que permitió que pudieran desarrollarse juegos con mucha más calidad que en las consolas de la generación anterior. Con la llegada de las consolas de quinta generación y sus procesadores de 32 y 64 bits, la calidad y complejidad de los juegos se disparó. **Saturn**<sup>[73]</sup> de Sega, **Nintendo 64**<sup>[74]</sup> de Nintendo y **PlayStation**<sup>[75]</sup> de Sony<sup>[76][77]</sup> fueron las más importantes, siendo esta última la que más éxito tuvo. Con esta generación se sustituyen los cartuchos por CDs<sup>[19]</sup> salvo en el caso de la Nintendo 64, cosa que repercutió negativamente en la compañía nipona.

Todos estos avances tanto en los ordenadores personales como en las consolas hicieron que los arcades se volvieran cada vez menos populares, ya que la potencia que ofrecían antiguamente dejó de ser superior a lo que podía tenerse en casa.

**Desde el año 2000** hasta hoy la industria de los videojuegos ha ido ganado en calidad, como cabría esperar viendo la evolución de las últimas décadas y los avances tecnológicos, pero posiblemente las novedades más significativas se aprecien en los cambios a la hora de concebir los juegos. Mientras que antes predominaban los juegos de un jugador o de varios pero con limitaciones (compartiendo el mismo ordenador o LAN<sup>[18]</sup>), en esta década el concepto de **juego online** se ha afianzado cada vez más. Ha sido notable la proliferación de los MMORPGs, pero también se le ha brindado la posibilidad de jugar online a otros videojuegos más orientados a un solo jugador.

Además de esto, pero también asociado a la mayor capacidad de compartir información gracias a Internet, se han hecho comunes los videojuegos que cuentan con la posibilidad de utilizar modificaciones hechas por el propio usuario. Esto ha sido especialmente notable en el caso del **Half-Life**<sup>[78]</sup> de Valve<sup>[79][80]</sup>, cuyo motor permitió lanzar en 1999 la modificación **Counter Strike**<sup>[81]</sup>, uno de los más famosos videojuegos de disparos en primera persona. **Baldur's Gate 2**<sup>[82]</sup> utiliza un sistema de scripts que permiten controlar muchos aspectos tanto de la historia del juego como del funcionamiento interno, lo cual ha hecho posible la creación de varias expansiones por parte de fans así como la corrección de muchos errores presentes en el juego. En el caso de juegos que utilizan mapas es muy común proporcionar un editor para crear nuevos, como ocurre con las series de **Heroes of Might and Magic** a partir del tercero, y es normal encontrar varias páginas web dedicadas a la distribución de estos desarrollos<sup>[83][84]</sup>.

De forma similar, muchos juegos permiten personalizar el aspecto visual del personaje. Esto se integra con el aspecto social que confiere a estos juegos la posibilidad del juego online, ya que se está interactuando constantemente con otros jugadores. En el caso de **Second Life**<sup>[85][86]</sup>, por ejemplo, abierto al público en junio de 2003, este posiblemente sea el aspecto más importante, ya que el juego carece de objetivos más que el de vivir un mundo virtual. Todo se centra en la interacción con el resto de jugadores, y la posibilidad de personalizar el mayor número de cosas posibles es esencial.

Esta proliferación de juegos online también se ha hecho visible en el mundo de las consolas. De la sexta generación en adelante, la mayoría de las consolas han incorporado la posibilidad de jugar online. La **Xbox**<sup>[87]</sup> de Microsoft<sup>[88][89]</sup>, de 2001, y la **PlayStation2**<sup>[90]</sup> de Sony, de 2000, son buenos ejemplos de esto. Pero no sólo la disponibilidad online ha sido un hito importante, también la inclusión de discos duros y sistemas operativos ha caracterizado la sexta generación de consolas. La pionera en esto fue Nintendo con el lanzamiento de su **GameCube**<sup>[91]</sup> en 2001, aunque no tuvo demasiado éxito en el mercado debido a la falta de juegos interesantes para la demanda de aquel momento.

En cuanto a la séptima generación, se caracteriza por consolas de mayor potencia de la mano de Microsoft y Sony, con sus respectivas **Xbox 360**<sup>[92]</sup> y **PlayStation 3**<sup>[93]</sup> (que además cuentan con sus propias tiendas online para poder comprar directamente desde la consola), y también por las innovaciones de Nintendo: la **Nintendo DS**<sup>[94]</sup> en 2004, la primera consola portátil con dos pantallas, una de ellas táctil, y la **Wii**<sup>[95]</sup> en 2006, un éxito gracias a su nuevo interfaz consistente en un mando con sensor de movimiento y control por infrarrojos que revolucionó la forma de juego convencional. Ambas consolas son menos potentes que las de Microsoft y Sony, pero Nintendo ha centrado su estrategia de ventas en la innovación que han supuesto.

Por último, cabe destacar la mejora de la calidad de los videojuegos en los teléfonos móviles, los cuales han pasado de ofrecer el **Snake** de Nokia a ser capaces de ejecutar títulos como **Monkey Island** y otros juegos similares. Muchos de los juegos actuales son versiones de otros más antiguos, tanto de ordenadores como de arcades, y es de esperar que su evolución sea similar a la que se ha ido observando en otro tipo de soportes.

## 6.2 *Los juegos online multijugador*

A lo largo de la historia de los videojuegos, los juegos online han adquirido cada vez más relevancia y protagonismo. Muchos de los juegos más sencillos de un jugador se adaptaron a este modelo con la proliferación de las pequeñas redes y posteriormente de Internet, llegando incluso a crearse juegos que no tienen forma de usarse offline, como los propios MMORPG.

Existen varias formas de jugar un videojuego online, algunas caracterizadas por la tecnología disponible en su época y otras por el tipo de juego en sí. A continuación se muestran las principales.

### 6.2.1 **Por E-Mail**

El origen de este modelo de juego surgió a partir de las partidas por correo ordinario<sup>[96]</sup>, en las que dos o más jugadores se mandan correspondencia indicando qué acciones toman en cada turno. Un ejemplo muy típico y que se lleva practicando durante cientos de años es el del **ajedrez**<sup>[97]</sup>, donde el jugador envía su movimiento al oponente y este lo replica en su propio tablero para luego contestar con su propio movimiento, y así sucesivamente.

Con la llegada de Internet, este formato de juego se comenzó a utilizar mediante el correo electrónico, con la ventaja de que se tarda mucho menos en jugar las partidas y es más sencillo organizarlas entre varios jugadores. Este sistema también se puede utilizar en juegos más complejos específicamente diseñados para ello, como **Age of Wonders**<sup>[98]</sup> o el famoso **VGA Planets**<sup>[99]</sup>, un juego basado en turnos donde es posible grabar la partida y enviársela a otros jugadores, que pueden cargarla y continuarla realizando sus movimientos.

Un grupo a destacar, ya que tiene que ver con este proyecto, es el de los juegos de rol por E-Mail, que permiten simular una partida cara a cara donde los jugadores pueden describir cómo son sus personajes y escribir ellos mismos la historia según se va desarrollando la partida. Las personas aficionadas a escribir pueden disfrutar mucho con

este tipo de juego, ya que además de jugar la partida pueden explayarse al describir escenarios y explicar las acciones y reacciones de su personaje. Es muy común jugar este tipo de partidas, más incluso que por correo electrónico, por foros, chats u otros canales de comunicación similares<sup>[100]</sup>.

Pese a que este modo de juego fue muy común hace un par de décadas, con la llegada de las nuevas tecnologías y todas las posibilidades que ofrecen se ha perdido el interés por jugar de esta forma, aunque todavía hay gente a quien le sigue gustando y páginas activas dedicadas a ello<sup>[101]</sup>.

### 6.2.2 Multi User Dungeons (MUDs)

Los MUDs<sup>[20][102]</sup> hoy en día son muy poco conocidos, pero son importantísimos a la hora de comprender los orígenes de los MMORPGS, ya que fueron los precursores de estos. No en vano la mecánica es la misma (mundos virtuales de fantasía donde interactúan varios jugadores de forma online y en tiempo real) pero se caracterizan porque la acción se lleva a cabo por texto. No tienen imágenes ni animaciones, las descripciones del entorno y los personajes son puro texto y para indicar las acciones a seguir se escriben comandos (por ejemplo, para mirar una fuente se escribe **mirar fuente**), como se puede observar en la *Imagen 2: Captura de pantalla de un MUD*.

```

Kthulu intenta golpear a Fedrah con su Baston pero falla.
Vida: xxx

Silend comienza a formular un hechizo.

Una gran zumbido se escucha en la zona. El zumbido va creciendo hasta que puedes
ver como una gran cantidad de abejas rabiosas que se aproximan!.

Silend pronuncia el cantico 'apex ovas summis'.

Las abejas envuelven a Gorey Di'Fum clavandoles su agujones.
Intentas golpear a Fedrah con tu Baston pero fallas.
Intentas alcanzar a Fedrah con tu Baston pero Esta retrocede quedando asf fuera de peligro.
Intentas golpear a Fedrah con tu Baston en su Cabeza de Minotauro pero no consigues calcular bien la distancia y fallas
el golpe.
Concentras tu pensamiento en toda abeja que se pueda encontrar en la zona para que venga en tu ayuda.

Una gran zumbido se escucha en la zona. El zumbido va creciendo hasta que puedes
ver como una gran cantidad de abejas rabiosas que se aproximan!.
Pronuncias el cantico 'apex ovas summis'.
Las abejas envuelven a Gorey Di'Fum clavandoles sus agujones.

Fedrah comienza a formular un hechizo.
fte 9
Comienzas a formular el hechizo 'Tormenta electrica'.
Clamas a los cielos por una tormenta: 'raytus zeus'! Los rayos empiezan a caer a tu alrededor como respuesta.

Una gran cantidad de rayos caen en la habitacion!
* Los 2 rayos de la tormenta van a caer sobre: Gorey Di'Fum y Fedrah.
>
Laelin Belgarath hiere gravemente a Gorey Di'Fum con su Cimitarra de la
Exhalaci6n abrasando su Amuleto del Trixkel.

Laelin Belgarath hiere gravemente a Gorey Di'Fum con su Espada de la oscuridad atraves6ndole su Capa de
Zhora.

Laelin Belgarath hiere gravemente a Gorey Di'Fum con su Cimitarra de la
Exhalaci6n abrasando su Cinturon de Quivira.

Laelin Belgarath hiere gravemente a Gorey Di'Fum con su Espada de la oscuridad atraves6ndole su
Esmeralda de los Suenos.

Laelin Belgarath hiere gravemente a Gorey Di'Fum con su Cimitarra de la
Exhalaci6n abrasando su Cabeza de Minotauro.

Laelin Belgarath hiere gravemente a Gorey Di'Fum con su Espada de la oscuridad atraves6ndole su
cuerpo.

Laelin Belgarath hiere gravemente a Gorey Di'Fum con su Cimitarra de la
Exhalaci6n abrasando su Pantal6n de Combate.

Gorey Di'Fum intenta golpear a Laelin Belgarath con su Maza de Fuego pero Esta
salta 6gilmente hacia un lado.

```

Imagen 2: Captura de pantalla de un MUD

Los orígenes de los MUDs se remontan al año 1975 con la creación de **Adventure**<sup>[103]</sup> por Will Crowther, el primer juego de aventuras por ordenador (en España se portó en 1989 con el nombre de La Aventura Original<sup>[104]</sup>). Era para un jugador, pero ya contenía muchos elementos de Dungeons & Dragons (D&D)<sup>[21] [105][106]</sup> y un sistema de juego típico de los MUDs. Posteriormente, en el verano de 1977 un grupo de estudiantes del MIT crearon **Zork**<sup>[107]</sup>, un juego inspirado en Adventure y que ganó mucha popularidad en ARPANET<sup>[22]</sup>. Este juego gustó tanto a Roy Trubshaw, un estudiante de la universidad de Essex (Inglaterra), que en 1978 comenzó a desarrollar **MUD**, después más conocido como **Essex MUD** o **MUD1**<sup>[108]</sup>, ahora sí con un sistema multijugador. En 1980 la universidad de Essex conectó su red interna con ARPANET, convirtiendo a MUD1 en el primer juego de rol multijugador de Internet.



Desde entonces los MUDs proliferaron sobre todo en ambientes universitarios, y se crearon una gran variedad de bases de código, cada una con sus propias características. La primera fue **AberMUD**<sup>[109]</sup>, escrita en 1987 por Alan Cox utilizando el lenguaje B<sup>[110]</sup>. A finales de 1988 el código fue portado a C<sup>[111]</sup>, donde se expandió rápidamente a muchas plataformas Unix<sup>[112]</sup> y se hizo muy popular, derivando en otras bases como **TinyMUD**<sup>[113]</sup>, **LPMUD**<sup>[114]</sup> y **DikuMUD**<sup>[115]</sup>, entre otras.

A continuación se muestra la *Imagen 3: Árbol de familia de DikuMUD*, donde se aprecia la evolución del MUD que más derivados ha tenido de los tres anteriormente mencionados.

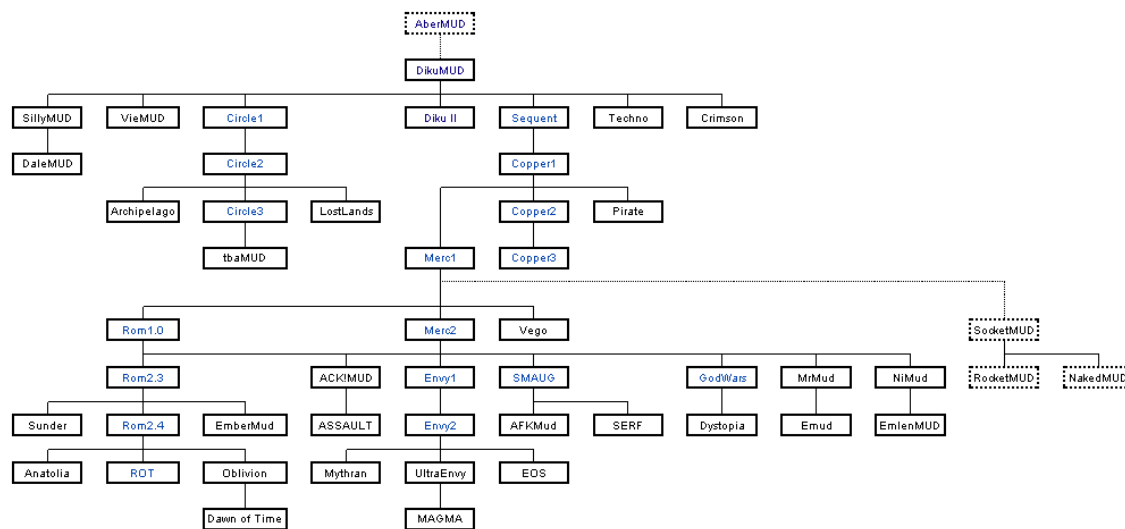


Imagen 3: Árbol de familia de DikuMUD

El tipo más común de MUD es probablemente el llamado **cortar y trinchar**, en el que el objetivo es simplemente crear un personaje para subirlo de nivel y obtener equipo matando monstruos. Sin embargo, hay muchos tipos más, como pueden ser los de **jugador contra jugador** (PvP<sup>[23]</sup>), también llamados PK<sup>[24]</sup>, en los que además de luchar contra monstruos los jugadores pueden combatir entre sí con más o menos restricciones; los orientados a **seguir un rol** (RP<sup>[25]</sup>), en donde el jugador debe centrarse en seguir el rol de su personaje para interactuar con el resto de jugadores; los **sociales**, que se centran en el contacto de los jugadores entre sí eliminando los aspectos más roleros del juego (personajes sin niveles, en universos sin monstruos, etc.); los **educacionales**, diseñados para ofrecer contenido educativo de algún tipo, e incluso los hay exclusivamente con **contenido adulto**.

A los MUD se accede generalmente utilizando cualquier cliente de Telnet o algún cliente específico para este tipo de juegos. El más conocido es el **CMUD**<sup>[116]</sup> (antiguamente zMUD) de la empresa **Zugg Software**<sup>[117]</sup>. Este tipo de clientes tienen muchas ventajas sobre el Telnet tradicional, ya que permiten entre otras cosas crear **macros** (asignación de acciones al pulsar una tecla), **disparadores** o triggers<sup>[26]</sup> (ejecución de acciones al recibir un texto determinado), programar **scripts**<sup>[27]</sup> (de forma que es posible incluso hacer que el personaje juegue solo), realizar **mapas** de forma automática e incluso reproducir sonidos en momentos determinados, ya sea porque el protocolo del propio MUD lo soporte o asociándolos a eventos usando los disparadores.

En España ha habido varios MUDs muy famosos, como por ejemplo **Illuvatar**, **Medina**, **Demon**, **Iluminado** o **Reinos de Leyenda 2**<sup>[118]</sup>, este último todavía activo. Hoy en día es raro encontrar MUDs en España debido a la proliferación de los MMORPG y al poco atractivo que despierta jugar en modo texto, pero sigue habiendo una gran cantidad de ellos funcionando a lo largo del mundo y páginas dedicadas a agruparlos<sup>[119]</sup>.

### 6.2.3 Primera persona

Este tipo de juegos utiliza una perspectiva en primera persona para mostrar la acción al jugador, de forma que se vea todo tal y como lo vería su personaje. Posiblemente el género más común de este tipo de juegos sea el de disparos (shooter), pero puede encontrarse en todo tipo de juegos, por ejemplo simuladores de vuelo. Tampoco es raro encontrar juegos que permitan utilizar varios tipos de perspectivas.

Se dice que el primer juego en primera persona es o **Spasim**<sup>[120]</sup> o **Maze War**<sup>[121]</sup>. No se sabe exactamente porque la fecha de Maze War no está bien documentada y su desarrollador no la recuerda, pero es cercana a la de Spasim, que salió a la luz en la primavera de 1974<sup>[122]</sup>.

Maze War pone al jugador dentro de un laberinto en el que puede moverse hacia adelante, atrás, derecha e izquierda. Originalmente para un jugador, poco después de su lanzamiento se le añadió la opción multijugador, en donde los jugadores podían

dispararse si se encontraban. Spasim es un simulador espacial multijugador con capacidad para 32 personas a través de la red universitaria PLATO<sup>[28][123]</sup>.

Poco a poco fueron surgiendo más juegos en primera persona, algunos sin modo multijugador, como **X-Wing**<sup>[29]</sup>, un simulador de combate espacial basado en Star Wars<sup>[124]</sup>, otros con modo multijugador sólo para LAN<sup>[18]</sup>, como los primeros de la serie **Marathon**, shooter de ciencia ficción originalmente creado para Macintosh<sup>[125]</sup>, y otros con modo multijugador a través de Internet, como **Unreal**<sup>[126]</sup>, otro shooter de ciencia ficción. El Ragnarok Online no dispone de modo en primera persona, pero otros MMORPG sí, como por ejemplo el **World of Warcraft**<sup>[127]</sup>, aunque no es obligatorio jugarlo así y puede optarse por la tercera persona.

## 6.2.4 Estrategia en tiempo real

Este tipo de género se caracteriza por tener generalmente una temática de juegos de guerra, en donde los jugadores deben derrotar a los ejércitos del enemigo en un entorno dinámico, sin turnos donde se puedan pensar los movimientos a seguir sin que los oponentes jueguen. Esto hace que haya que la toma de decisiones y su ejecución deba ser muy rápida, ya que el tiempo que se pierda en pensar los próximos movimientos lo puede emplear el oponente para desarrollar su estrategia.

**Utopia**<sup>[128]</sup>, desarrollado por Don Daglow en 1982 para la consola Intellivision, puede considerarse el precursor de los videojuegos en tiempo real. En este juego, dos jugadores humanos compiten para desarrollar su isla construyendo edificios, barcos y creando actividades rebeldes en la isla del oponente. También es posible destruir los barcos del contrincante. El jugador que más puntos tiene al final de un número de turnos establecido, es el ganador.

En 1989 Sega publicó **Herzog Zwei**<sup>[129]</sup> para su consola Mega Drive. Este juego presentaba características más comunes a los juegos de estrategia en tiempo real modernos, ya que a pesar de controlar una sola unidad, la forma de manejarla era la que se utiliza hoy en día de hacer clic en el punto donde se quiere dirigir. También era de destacar el control y destrucción de bases, al igual que la economía y producción que proporcionaban.

Cabe destacar el videojuego **StarCraft**<sup>[130]</sup> de Blizzard Entertainment<sup>[131]</sup>, un clásico de estrategia en tiempo real de ambientación futurista que ha sido y sigue siendo ampliamente jugado desde su aparición en 1998. Además hace uso de **Battle.net**<sup>[132]</sup>, un servicio para partidas online multijugador también de Blizzard que revolucionó esta forma de jugar, ya que no es necesario salir del juego y buscar servidores para poder unirse a una partida (no en vano la idea es ahora utilizada en la gran mayoría de los juegos online de distintas empresas).

La ventaja que tienen los juegos de estrategia en tiempo real frente a los de estrategia por turnos<sup>[30]</sup> en las partidas multijugador es que se puede jugar de forma dinámica sin tener que esperar a que los oponentes finalicen sus turnos, algo que puede alargar mucho las partidas dependiendo de la complejidad del juego y la implicación de los demás jugadores. Incluso en juegos sencillos, los turnos siempre hacen la partida más lenta y más propensa a distraerse haciendo otras cosas mientras el oponente finaliza su turno, lo que en muchos casos la alarga más. Para intentar solucionar esto, algunos juegos como el **Heroes of Might and Magic V**<sup>[31][133]</sup> de Ubisoft<sup>[134]</sup> han añadido a las partidas multijugador componentes gracias a los cuales se pueden realizar ciertas acciones durante los turnos de los oponentes.

### 6.2.5 Por navegador

Los juegos por navegador son muy típicos desde hace unos años. Tienen la ventaja de que no es necesario instalar un software adicional, por lo que es muy rápido empezar a jugar ya que no es necesario comprar o descargar ningún programa, y puede jugarse desde prácticamente cualquier ordenador. Es común que estos juegos se desarrollen utilizando tecnología Java<sup>[32][135]</sup>, Flash<sup>[33]</sup> o similar.

Los juegos multijugador por navegador pueden ser de distintos tipos, con una variedad cada vez mayor según avanza la tecnología. Son comunes y muy jugados los juegos por turnos o en tiempo real como OGame<sup>[136]</sup> o Travian<sup>[137]</sup>. Normalmente cada jugador tiene una serie de turnos para realizar ciertas acciones que los gastan, y estos se regeneran de forma automática con el paso de los minutos o las horas. En el caso de los juegos en tiempo real, cada acción tarda un cierto tiempo en llevarse a cabo y suelen ser

necesarios recursos que también tardan un tiempo en generarse. Estos sistemas hacen que se pueda jugar dedicando poco tiempo al día pero permiten desarrollar distintas estrategias, sobre todo interactuando con el resto de jugadores mediante sistemas de alianzas.

También existen muchos juegos casuales de distintos tipos, desde simples **Tetris**<sup>[138][139]</sup> hasta juegos de **coordinación**<sup>[140]</sup> ofrecidos por las Fuerzas Armadas Suecas. Este tipo de juegos no suele tener un fin en concreto ni precisa de ninguna continuidad, y son muy comunes también sus contrapartidas de un solo jugador, probablemente más que las de multijugador.

### 6.2.6 Massive Multiplayer Online Role-Playing Games (MMORPGs)

El Ragnarok Online, y por tanto el servidor BidaiRO, entra dentro de la categoría de juego de rol multijugador online masivo. En este género de juegos un gran número de jugadores interactúan los unos con los otros en un universo virtual, ya sea para realizar acciones conjuntas como para combatir entre ellos. Como en todo juego de rol, el jugador asume el rol de un personaje de ficción (PJ<sup>[34]</sup>) que él mismo puede personalizar de forma más o menos libre, dependiendo de las características que ofrezca el juego, y jugarlo en un mundo normalmente cambiante gracias a las actualizaciones que las compañías propietarias de estos juegos lanzan periódicamente.

El término MMORPG fue acuñado por Richard Garriott, el creador de **Ultima Online**<sup>[141]</sup> en 1997. Antes de eso, a los videojuegos con estas características (e incluso al propio Ultima Online durante un tiempo) se los llamaba **MUDs gráficos**. Ultima Online, pese a tener un entorno gráfico avanzado para la época, todavía conservaba características propias de los MUDs en forma de información mediante texto. No era necesario escribir las acciones que querían realizarse, pero sí se recibía un constante feedback a través de una consola de texto. El primer RPG<sup>[35]</sup> totalmente gráfico fue **Neverwinter Nights**<sup>[142]</sup> de AOL, de 1991 (no confundir con el más reciente Neverwinter Nights de BioWare), pero el primer MMORPG no surgió hasta 1996 bajo el nombre de **Meridian 59**<sup>[143]</sup>.

Normalmente los universos de los MMORPGS están basados en mundos de fantasía al estilo D&D, pero también los hay de ciencia ficción, ambientados en otras épocas o incluso una mezcla de varios estilos. El Ragnarok Online, por ejemplo, comenzó ambientándose en la mitología nórdica con personajes al estilo D&D, y hoy en día presenta una mezcla aún mayor de temáticas donde se incluyen armas de fuego, zonas de ambientación japonesa, futurista, e incluso puede encontrarse la fábrica de juguetes de Papá Noel.

Generalmente el objetivo de un MMORPG es la progresión del personaje a base de subir niveles<sup>[36]</sup> mediante la ganancia de puntos de experiencia<sup>[37]</sup>, y la obtención de equipo y habilidades extra. Esto suele ser un proceso repetitivo, sobre todo cuando se alcanzan niveles superiores, por lo que a la larga es motivo de tedio para muchos jugadores.

Estos objetivos pueden conseguirse en solitario o con la ayuda de otros jugadores. Normalmente en un MMORPG se crearán agrupaciones de usuarios con intereses comunes, lo que fomenta la socialización en el juego. Este es un punto importante puesto que en muchos de estos juegos hay objetivos que es imposible conseguir en solitario, por lo que es necesaria la ayuda de más jugadores. En el World of Warcraft, por ejemplo, hay partes del juego a las que sólo puede accederse si se va en grupo. Recientemente se han incorporado zonas así en el Ragnarok Online, aunque en comparación tiene menos; aun así hay muchas otras en las que se hace necesario ir en grupo para poder lograr algo, aunque no sea obligatorio.

La mayoría de los MMORPG siguen una arquitectura cliente-servidor<sup>[38]</sup>, en la que el mundo virtual corre en el servidor y los jugadores se conectan a él mediante un cliente. Normalmente es el cliente el que proporciona la parte pesada del juego, es decir, los gráficos, sonidos, animaciones, etc., de forma que se reduce drásticamente el ancho de banda necesario para jugar. También es el que se actualiza periódicamente para proporcionar acceso a nuevos contenidos, aunque la parte del servidor también debe actualizarse. Este es el sistema que utiliza el Ragnarok Online.

Normalmente la cuenta de un MMORPG requiere una cuota mensual y da derecho a varios PJs, como es el caso del Ragnarok Online. Esto es necesario para

mantener los costes de ofrecer un servicio online y mantenerlo actualizado, aunque existen otros modelos de negocio, como por ejemplo el de **Guild Wars**<sup>[144]</sup>, en el que se paga una única cuota para obtener el juego básico y luego es posible comprar packs de expansión<sup>[39]</sup>. Un modelo similar es el que utilizan los MMORPG “gratuitos”, en los que es posible jugar sin pagar nada pero ofrecen la posibilidad de comprar mejoras estéticas, equipo o cualquier otro tipo de ventajas. Este tipo de modelo es muy utilizado en Corea, de donde proviene el Ragnarok Online (de hecho mezcla este sistema con el de cuotas mensuales).

Normalmente un MMORPG no estará constituido sólo por un servidor, sino que tendrá varios para poder repartir de forma eficiente la carga que supone el gran número de usuarios que se conectan. Estos servidores son independientes entre sí, de forma que un PJ sólo podrá usarse en el servidor en el que se ha creado su cuenta y nada de lo que ocurra en otros servidores le afectará. Esto hace que en muchos casos las características entre servidores de un mismo juego difieran, y es especialmente notable en el Ragnarok Online, donde entre sus propios servidores hay reglas ligeramente diferentes, objetos distintos, características únicas para cada servidor e incluso las actualizaciones no están sincronizadas. El servidor coreano es siempre el primero en actualizarse, y las actualizaciones pueden tardar incluso un año en llegar al europeo.

Los MMORPG suelen ser bastante adictivos, ya que mezclan un universo siempre cambiante con el afán de hacer crecer al PJ, y generalmente la variedad de personajes a elegir es suficiente para adaptarse a las preferencias de la mayoría de los jugadores. Mientras haya metas y objetivos que alcanzar, los jugadores estarán contentos con el juego y querrán seguir jugando para lograr esas metas. Es por esto que al alcanzar los niveles más altos, donde cada vez quedan menos cosas por hacer y es más difícil notar el progreso, muchos jugadores comienzan a perder interés por el juego. Por eso una buena estrategia de las compañías es ofrecer actualizaciones y expansiones que permitan llegar más allá del límite establecido previamente, de forma que vuelva a crecer el interés de los jugadores más avanzados.

## 6.3 *Ragnarok Online*

A continuación se muestran los aspectos más básicos del Ragnarok, que ayudarán a comprender el tipo de juego que es y a familiarizarse con los términos y eventos más importantes.

El Ragnarok cuenta también con una historia que puede descubrirse realizando multitud de aventuras en las que a medida que el jugador va resolviendo distintas pruebas, se le va informando de los orígenes y desarrollo del universo. Aunque son historias muy completas, generalmente los jugadores las desconocen ya que se limitan a completar los objetivos sin prestar atención a la propia historia (esto es posible debido a las múltiples guías<sup>[145]</sup> disponibles en Internet, donde no hay más que seguir los pasos que describen para completar las pruebas).

### 6.3.1 Mecánica del juego

En el Ragnarok Online, el jugador crea uno o más personajes (normalmente uno principal y varios secundarios o de soporte) y se dedica a subirlo de nivel, obteniendo cada vez más equipo y habilidades en el proceso. No tiene una meta en concreto, cada jugador puede tener la suya propia, aunque lo más normal es que los jugadores se junten en clanes<sup>[40]</sup> y luchen contra otros por obtener uno o más castillos, que proporcionan prestigio y objetos para los miembros del clan. También hay clanes que prefieren dedicarse a la exploración del mundo virtual y buscan otro tipo de metas.

El mundo del Ragnarok está dividido en mapas, cada uno con su propia ambientación y monstruos, aunque puede encontrarse un mismo tipo de monstruos en más de un mapa. Para cambiar de un mapa a otro se utilizan los warps<sup>[41]</sup>, cosa que requiere un proceso breve de carga. Los monstruos no pueden viajar entre mapas.

Hay cuatro naciones principales en el mundo del Ragnarok: **Rune Midgard**, la capital del reino que da nombre al continente, la **República Schwartzvalt**, una zona industrializada al norte de Rune Midgard, **Arunafeltz**, una nación religiosa ambientada



en Israel y Turquía y **Ash Vacuum** o el Nuevo Mundo, una nueva tierra descubierta en otro continente.

En la *Imagen 4: Continente Rune Midgard*, se puede observar el mapa del continente con Rune Midgard ocupando la zona central y sur, República Schwartzvalt al norte y Arunafeltz al noroeste. Ash Vacuum, así como algunas otras zonas, no aparecen en este mapa.



**Imagen 4: Continente Rune Midgard**

### **6.3.2 Atributos del personaje**

A la hora de crear un personaje, el jugador debe tener en cuenta varios factores que influirán en cómo será de competente al realizar cada tipo de acción. Para ello se utilizan los atributos o stats<sup>[42]</sup>, altamente personalizables y que hacen del Ragnarok un juego muy flexible, en el que un mismo tipo de personaje puede ser muy diferente dependiendo de cómo lo haya decidido enfocar el jugador.

Cada vez que se sube un nivel, el jugador obtiene un cierto número de puntos de stats que puede gastar en subir estos atributos. Dependiendo del nivel obtendrá más o menos puntos, y cuanto más alto sea el stat que quiera subir, más puntos requerirá, por

lo que un stat bajo subirá mucho más rápido que uno alto, pero también será menos efectivo.

Hay seis stats que el jugador puede subir a su antojo, siempre que tenga puntos suficientes:

- **Fuerza (STR):** Mide la fuerza física del personaje e influye en el daño que hacen sus ataques cuerpo a cuerpo, y en menor medida en los ataques de rango. También influye en el peso total que puede cargar.
- **Agilidad (AGI):** Indica lo ágil que es el personaje, repercutiendo en la velocidad a la cual puede atacar y la capacidad de esquivar (flee<sup>[43]</sup>) ataques físicos.
- **Vitalidad (VIT):** La vitalidad afecta al número de puntos de vida o HP<sup>[44]</sup> que puede llegar a tener el personaje, así como a su defensa física y resistencia natural a muchos efectos negativos. También incrementa la efectividad de los ítems de curación<sup>[45]</sup> y la regeneración de HP<sup>[46]</sup>.
- **Inteligencia (INT):** Aumenta el daño mágico que puede realizar el personaje, y de forma similar a la vitalidad, afecta al número de puntos de habilidad o SP<sup>[47]</sup> que puede llegar a tener, así como a su defensa mágica y la resistencia natural a algunos efectos negativos distintos de los que protege la vitalidad. También incrementa la efectividad de los ítems de recuperación de SP<sup>[48]</sup> y la regeneración de SP<sup>[49]</sup>.
- **Destreza (DEX):** La destreza aumenta el daño realizado con ataques físicos de rango, determina la capacidad del jugador de golpear con éxito a un objetivo y reduce el tiempo necesario para utilizar una habilidad<sup>[50]</sup>.
- **Suerte (LUK):** Incrementa la probabilidad de ejecutar un golpe crítico<sup>[51]</sup>, así como la posibilidad de esquivar un golpe completamente (perfect dodge<sup>[52]</sup>). También reduce la probabilidad de ser golpeado con un crítico y ayuda ligeramente a resistir algunos efectos negativos.

Además de estos stats, hay otros que el jugador no puede subir manualmente pero que dependen de los explicados anteriormente:

- **ATK:** Es el daño base que hará el personaje al realizar un ataque físico, depende principalmente de la fuerza y del arma equipada.
- **MATK:** Es el daño base que hará el personaje al realizar un ataque mágico, depende principalmente de la inteligencia y del arma equipada.
- **Flee:** Podría traducirse como *esquivar*, indica la capacidad del personaje de evitar un golpe físico, y depende principalmente de su agilidad. El valor es orientativo, puesto que también influye la capacidad de golpear del oponente.
- **Hit:** Del inglés *golpe*, es la capacidad del personaje para golpear a otro, y depende principalmente de su destreza. El valor es orientativo, puesto que influye la capacidad de esquivar del oponente.
- **Critical:** Es la probabilidad de realizar un golpe crítico al oponente, y depende principalmente de la suerte. También es orientativo puesto que puede reducirse dependiendo de la suerte del oponente.
- **Def:** La defensa del personaje. Se muestra con dos números de la forma  $X + Y$ , donde  $X$  es la defensa que proporciona el equipo e indica el porcentaje en el que se reducirá el daño recibido, e  $Y$  es la defensa obtenida por la vitalidad del personaje, la cual se resta directamente del daño recibido.
- **Mdef:** Funciona exactamente igual que la def, pero representa la defensa frente a ataques mágicos.
- **ASPD:** Acrónimo de *attack speed* (velocidad de ataque), indica la rapidez con la que se realizan los ataques físicos, y depende principalmente de la agilidad y en menor medida de la destreza.
- **Perfect Dodge:** Es la probabilidad de esquivar completamente un golpe físico, independientemente del hit del atacante, y depende principalmente de la suerte.

Todos estos stats pueden ser modificados por equipo y distintas habilidades. En la *Imagen 5: Ventana de stats y equipo*, se observa la ventana donde se pueden consultar los atributos descritos anteriormente, junto con el equipo que se está usando.



Imagen 5: Ventana de stats y equipo

### 6.3.3 Clases

Las clases, o jobs<sup>[53]</sup>, es el aspecto más importante del personaje, ya que es lo que va a definir sus puntos fuertes y sus debilidades, y el rol que desempeñará en el juego tanto de forma individual como en grupo. Casi la totalidad de este tipo de juegos contempla jobs, por ejemplo en la primera edición de D&D están los magos, ladrones, guerreros y clérigos. En el Ragnarok la mayoría de los jobs derivan de estos cuatro, pero hay algunos más.

El juego inicialmente estaba compuesto de 13 jobs, pero actualmente dispone de 50 que han ido añadiéndose mediante actualizaciones, y posiblemente implementen alguno más en un futuro. Cada job dispone de ciertas habilidades exclusivas y obtiene distintas bonificaciones a los stats. Además, casi la totalidad del equipo está restringido por jobs, por lo que según el job escogido se podrán equipar unas cosas u otras.

El jugador, además de ganar experiencia para subir de nivel, gana también experiencia para subir el nivel del job<sup>[54]</sup>. Por cada nivel de job se obtiene un **punto de habilidad**<sup>[55]</sup> (no confundir con el SP descrito anteriormente), que podrá canjearse para

obtener o mejorar habilidades propias de ese job, y se obtienen mejores bonificaciones a los stats.

El job inicial de todo jugador es el de *novato* (novice), y una vez alcanzado el nivel de job 9 es posible cambiar de clase, pudiendo elegir entre 9 jobs más. De esos 9, 7 son **primeros jobs**<sup>[56]</sup> que permiten cambiar de nuevo de clase una vez alcanzado un nivel mínimo de job de 40 (con un máximo de 50). Se podrá elegir entre 2 jobs más que dependerán de la clase primaria que se haya escogido previamente. Estos son los **segundos jobs**<sup>[57]</sup>.

Por ejemplo, un personaje puede comenzar siendo un novice y en el nivel 9 cambiar de job a espadachín (swordman). Una vez alcanzado el nivel 40 de swordman, puede elegir entre cambiar a caballero (knight) o a cruzado (crussader). Ambos jobs derivan del mismo y comparten características similares (son jobs orientados a combate cuerpo a cuerpo), pero el knight tendrá más habilidades ofensivas mientras que el crussader las tendrá defensivas.

Tras alcanzar el nivel 99 de base y 50 de segundo job, el jugador puede **renacer**<sup>[58]</sup> al personaje, lo que implica volver a empezar desde nivel 1 y perder todos los jobs, que podrán volver a obtenerse subiendo niveles. A estos nuevos jobs se les llama **extendidos**<sup>[59]</sup>, y las segundas clases extendidas poseen más habilidades que las anteriores, mejores bonificaciones a los stats y pueden subirse hasta un máximo de 70 niveles.

Recientemente, los servidores coreanos han añadido los **terceros jobs**<sup>[60]</sup>, que al igual que ocurre con los segundos jobs, son una mejora de estas con más habilidades. Es posible cambiar al tercer job correspondiente una vez alcanzado el nivel 50 de segundo job sin renacer, o el nivel 70 de segundo job habiendo renacido, lo cual es más ventajoso pero mucho más lento.

A continuación se describen brevemente los primeros jobs. Se utilizan siempre los nombres en inglés, al menos entre los jugadores de habla hispana, por lo que se respetará ese uso en este documento, pero se puede encontrar una traducción aproximada en el *Anexo A: Traducciones de las clases del Ragnarok*:

- **Swordman**: Especialista en combate cuerpo a cuerpo con armas.
- **Magician**: Especialista en el uso de la magia elemental.
- **Archer**: Especialista en ataques a distancia con arcos.
- **Merchant**: Especialista en comercio.
- **Acolyte**: Especialista en dar soporte a otros jugadores.
- **Thief**: Especialista en sigilo, robos y ataques con un arma en cada mano.
- **Taekwon**: Especialista en Taekwondo.
- **Gunslinger**: Especialista en armas de fuego (no tiene segundo job).
- **Ninja**: Especialista en magia elemental, sigilo, shurikens y dagas (no tiene segundo job).

A continuación se describen brevemente los segundos jobs con sus correspondientes jobs extendidos, si aplica:

- **Super novice**: Especializado en todas las habilidades de los primeros jobs (no puede renacer).
- **Knight/Lord knight**: Especializado en lanzas y espadas a dos manos, puede llevar montura y tiene una gran resistencia y velocidad de ataque.
- **Crusader/Paladin**: Especializado en espada y escudo y lanzas a dos manos, puede llevar montura y tiene una gran resistencia además de habilidades de soporte.
- **Assassin/Assassin cross**: Especializado en dagas y katars, sigilo, uso de venenos y ataques muy rápidos y dañinos.
- **Rogue/Stalker**: Especializado en sigilo (más que los asesinos), copiar habilidades de otros jobs e inutilizar personajes.
- **Priest/High priest**: Especializado en dar soporte, curación, exorcismos y algunas maldiciones.

- **Monk/Champion:** Especializado en ataques de puño o con mazas para realizar daño masivo y un poco de soporte y curación.
- **Wizard/High wizard:** Especializado en magia elemental de área.
- **Sage/Professor:** Especializado en magia elemental, soporte y efectos negativos contra sus enemigos.
- **Blacksmith/Whitesmith:** Especializado en forja de armas y en daño masivo utilizando dinero.
- **Alchemist/Creator:** Especializado en pociones tanto de ataque como defensa y en la creación de vida.
- **Hunter/Sniper:** Especializado en ataques a distancia con arcos y en el uso de trampas.
- **Bard/Clown:** Especializado en el uso de instrumentos musicales, habilidades de soporte y alguna de inhabilitación.
- **Dancer/Gipsy:** Especializada en el uso de látigos, habilidades de inhabilitación y alguna de soporte.
- **Soul linker:** Proporciona habilidades especiales a cada job (no puede renacer).
- **Star Gladiator:** Especializado en causar grandes cantidades de daño a ciertos tipos de enemigos en determinados días y mapas (no puede renacer).

A continuación se describen brevemente los terceros jobs:

- **Arc bishop:** Habilidades de soporte masivas y aumentados los exorcismos y ataques directos.
- **Gulliotine cross:** Incrementadas las habilidades con venenos.
- **Rune knight:** Habilidades mágicas mediante runas y posibilidad de montar y atacar con un dragón.
- **Mechanic:** Especializado en robótica, dispone de una armadura mágico-tecnológica con habilidades de ataque y defensa.
- **Ranger:** Incrementadas las habilidades con las trampas y posibilidad de montar y ser ayudado por un lobo.

- **Warlock:** Incrementado el daño mágico y el área de efecto de sus conjuros.
- **Shadow chaser:** Incrementada su habilidad para copiar otras habilidades y maestría en grafitis y máscaras para inhabilitar oponentes.
- **Wanderer:** Habilidades para incrementar el daño y la defensa mágica y aumento de sus habilidades ofensivas.
- **Shura:** Incrementados los ataques físicos de tipo combo, más habilidades espirituales y de apoyo.
- **Royal guard:** Incrementadas las habilidades de lanza y añadidas habilidades grupales, además de la posibilidad de montar un grifo<sup>[61]</sup>.
- **Sorcerer:** Añadidos conjuros elementales de área, invocaciones y habilidades de soporte.
- **Genetic:** Aumentada la capacidad de crear distintos tipos de pociones, creación de nuevas formas de vida.
- **Minstrel:** Habilidades para incrementar el daño y la defensa física y aumento de sus habilidades ofensivas.

En la *Imagen 6: Tabla con los distintos jobs*, se puede apreciar de forma visual la distribución de las clases.



Novice Class																			
Novice																			
Super Novice	First class																		
	Swordman		Acolyte		Mage		Thief		Merchant		Archer		Taekwon Kid		Star Gladiator  (Taekwon Master)	Soul Linker	Ninja	Gunslinger	
	Second class																		
	Knight	Crusader	Priest	Monk	Wizard	Sage	Assassin	Rogue	Blacksmith	Alchemist	Hunter	Bard / Dancer							
	Transcendent																		
	High Novice																		
	Transcendent first class																		
	High Swordman		High Acolyte		High Mage		High Thief		High Merchant		High Archer								
	Transcendent second class																		
	Lord Knight	Paladin	High Priest	Champion	High Wizard	Professor (Scholar)	Assassin Cross	Stalker	Whitesmith (Mastersmith)	Creator (Biochemist)	Sniper	Clown / Gypsy							
Third Class																			
Rune Knight	Royal Guard	Arc Bishop	Shura	Warlock	Sorcerer	Guillotine Cross	Shadow Chaser	Mechanic	Genetic	Ranger	Minstrel / Wanderer								

### 6.3.4 Clanes

Los clanes, del inglés *guild*, componen uno de los aspectos sociales más importantes del juego. Es la forma que tienen los jugadores de reunirse bajo una misma bandera para compartir objetivos comunes o simplemente para jugar acompañados. Generalmente se establece un vínculo más fuerte entre miembros de un mismo clan que entre el resto de jugadores, ya que con los miembros del clan se puede charlar en cualquier momento mientras se está jugando, aunque no estén ellos cerca. En los servidores oficiales la única forma de comunicarse con la gente que no está cerca del jugador, aparte de usar el chat de guild, es mediante susurros (whisp), que viene a ser un canal directo entre el propio jugador y el personaje con el que se quiere comunicar, pero está limitado a un tipo de comunicación uno a uno y privada. Para fomentar la comunicación entre jugadores, en BidaiRO se ha implementado un chat global<sup>[62]</sup> de forma que se puede hablar en público con todos los jugadores.

Cada miembro del clan llevará un pequeño emblema al lado de su personaje, de forma que es fácilmente reconocible a qué clan pertenece cada uno. También, junto a su nombre, llevará el título<sup>[63]</sup> del clan, que viene a ser un rango asignado por el Señor del Clan (Guild Master o GM<sup>[64]</sup>). El GM es el miembro fundador del clan, e inicialmente se encarga de elegir el emblema que identificará a la guild, incluir a nuevos miembros y crear una lista de títulos y sus tasas<sup>[65]</sup>.

El clan, al igual que los jugadores, puede subir de nivel, y la experiencia necesaria para ello se obtiene de un porcentaje de la experiencia base de los miembros, según la tasa que tengan. Los beneficios de subir de nivel es la obtención de un punto de habilidad para gastarlo en habilidades de clan, que afectarán a distintos aspectos de este (aumento de la capacidad máxima, posibilidad de invocar a los miembros, aumento de stats cuando se está cerca del GM, etc.). Sólo el GM puede elegir qué skills obtener y sólo él puede utilizarlas, en caso de que sean activas<sup>[66]</sup>.

El GM también puede otorgar privilegios de inclusión y/o expulsión de miembros a los títulos que quiera (es decir, al igual que las tasas, no es de forma individual), pero es lo único que puede delegar.

En la *Imagen 7: Ventana de guild y personaje con el emblema junto a su nombre*, se puede observar la ventana de información del clan con la pestaña principal. Cada pestaña muestra un tipo de información diferente.



*Imagen 7: Ventana de guild y personaje con el emblema junto a su nombre*

Uno de los objetivos más comunes que buscan los miembros de un mismo clan es luchar contra **MVPs** (del inglés *Most Valuable Player*, jugador más valioso), monstruos muy poderosos que sólo pueden ser derrotados en grupo y suelen otorgar objetos valiosos y una gran cantidad de experiencia. El nombre MVP, que realmente hace referencia a un jugador y no al monstruo en sí, viene del símbolo que aparece sobre la cabeza del personaje que ha realizado más daño cuando el monstruo muere.

### 6.3.5 Guerra del Emperium

Más comúnmente conocidas por su nombre en inglés, las **War of Emperium** (WoE<sup>[67]</sup>) consisten en una lucha entre clanes por conquistar uno o más de los castillos que se encuentran repartidos en las zonas cercanas a algunas de las ciudades del juego. El clan que consiga conquistar y mantener uno o más castillos ganará prestigio y obtendrá varios beneficios.

Las WoE duran dos horas en los servidores oficiales (en BidaiRO se ha reducido a una) y se celebran dos veces por semana. Para conquistar un castillo, es necesario

recorrer sus habitaciones hasta llegar a la sala del **Emperium**, que no es más que una piedra estática que hay que destruir. El clan que lo destruya pasará a ser el nuevo dueño del castillo, expulsando a los jugadores que no pertenezcan a ese clan, que deberán volver al castillo y tratar de destruir nuevamente al Emperium, que se regenera cada vez que se destruye. Es posible realizar alianzas formales entre clanes para tener más posibilidades de conquistar los castillos.

Cuando las WoE finalizan, los GM de los clanes que tengan algún castillo podrán gestionarlos invirtiendo en defensa, economía y comprando guardianes que ayudarán en su defensa. Las inversiones en defensa sirven para aumentar la resistencia de los guardianes y del Emperium, y la de economía hace que el castillo produzca más cofres. Estos cofres se generan cada día y en su interior pueden encontrarse varios objetos beneficiosos para los miembros del clan. Además, aunque todos los castillos ofrecen una base común de objetos, cada castillo tiene algún objeto único que sólo puede obtenerse conquistándolo.

Otra ventaja de tener un castillo es que se abre acceso a las **Mazmorras de Clan** (*Guild Dungeons*), a las que sólo tienen acceso los miembros de la guild propietaria del castillo. Básicamente se trata de mapas con algunos monstruos únicos, con la peculiaridad de que hay menos mazmorras que castillos, por lo que son compartidas y es posible encontrar miembros de otras guilds que posean otro castillo común. En estos mapas es posible atacarse entre guilds enemigas, por lo que es un factor a tener en cuenta.

Inicialmente sólo existía un tipo de WoE, pero en 2008 se implementaron las **WoE SE** (*segunda edición*) o **WoE 2.0**. El funcionamiento es similar a las primeras, pero los castillos son mucho más amplios y complejos. Además tienen guardianes más fuertes y es más complicado llegar al Emperium: hay que destruir previamente dos piedras guardianas y después tres barricadas, haciendo que las estrategias sean muy diferentes y se requiera una buena coordinación en los ataques. Estas WoE se juegan en 10 castillos frente a los 20 de las originales, que tienen sus propios cofres y mazmorras.

Las WoE están pensadas para servidores de miles de personas, de forma que en BidaiRO, siendo un servidor pequeño, sólo están activadas las WoE 1.0 y sólo se utiliza un castillo, que se cambia cada mes para proporcionar más variedad.

En la *Imagen 8: Un castillo con un Emperium*, se muestra la sala del Emperium de un castillo vacío, antes del comienzo de una WoE.



**Imagen 8: Un castillo con un Emperium**

### **6.3.6 Futuro del Ragnarok Online**

Pese a ser un MMORPG con un sistema de juego muy bueno, donde las características del personaje son mucho más personalizables que en la mayoría de juegos de este tipo, el RO cada vez cuenta con menos jugadores, al menos en el ámbito hispano. Al juego ya se le puede considerar viejo, y esto se nota sobre todo en los gráficos: el Ragnarok es una mezcla de gráficos en 3D (los mapas) con gráficos en 2D (todo lo demás), por lo que no puede competir con otros MMORPG realizados

completamente en 3D y que hacen especial énfasis en los efectos visuales. No es raro que muchos antiguos jugadores de RO lo hayan dejado por el World of Warcraft, por ejemplo.

En Corea y alrededores, sin embargo, parece que todavía es un juego bastante importante. Se dice que lo que el World of Warcraft es para Europa, el Ragnarok Online lo es para Corea. Las constantes actualizaciones realizadas por el equipo coreano, que luego son implementadas en el resto de países, parecen corroborar esto. No sólo actualizaciones de mapas y objetos, sino del propio sistema de juego, algunas muy complejas que han llegado a cambiar completamente muchos aspectos importantes de este<sup>[146]</sup> (cosa que ha gustado mucho por una parte, ya que mejora muchos aspectos del juego, pero por otra ha sido objeto de grandes críticas, ya que los jugadores suelen modelar a sus personajes basándose en un sistema de juego determinado, y si este cambia el personaje puede quedar inservible).

Gravity también lleva años trabajando en la segunda parte de Ragnarok Online. En 2006 comenzó la beta cerrada<sup>[68]</sup> en Corea de **Ragnarok Online 2: The Gate of the World**<sup>[147]</sup>, en 3D y con un sistema de juego diferente, que se prolongó hasta 2010, momento en el que se realizó una beta abierta. Se puede observar su aspecto en la *Imagen 9: Ragnarok Online 2: The Gate of the World*. Sin embargo, el juego tuvo una muy mala acogida, por lo que sufrió muchos cambios durante estas pruebas. En julio de 2010 se anunció el abandono del desarrollo y la creación de una nueva versión, **Ragnarok Online 2: Legend of the Second**<sup>[148]</sup>, que mantendrá la mecánica del Ragnarok original pero seguirá utilizando los gráficos en 3D. La beta cerrada está prevista para el 31 de agosto de 2010, y es de suponer que gustará más que el anterior intento, ya que mantendrá un buen sistema de juego añadiéndole una calidad gráfica superior.

Es de esperar por tanto que, al menos en lo que respecta a Europa, América Latina y posiblemente Estados Unidos, cada vez haya menos jugadores de Ragnarok. En España casi todos son jugadores veteranos que se mantienen fieles al juego, pero es difícil que alguien que no haya jugado antes se decante por este MMORPG teniendo otros mucho más atractivos visualmente. Generalmente los nuevos jugadores son amigos de los que ya juegan, o aficionados al manga<sup>[69]</sup> que se sienten atraídos por la

similitud gráfica del Ragnarok con estos cómics. Quizá con la salida del Ragnarok 2, si los gráficos son lo suficientemente buenos, se vuelva a disparar el interés por este juego, pero hasta entonces sus usuarios irán en descenso.



Imagen 9: Ragnarok Online 2: The Gate of the World

## 6.4 Servidores de Ragnarok Online

### 6.4.1 Oficiales

Los servidores oficiales de Gravity están repartidos por zonas geográficas, y se resumen en la *Tabla 1: Servidores oficiales de RO*<sup>[149]</sup>:

Localización	Acrónimo
<b>Brasil</b>	bRO
<b>Europa</b>	euRO
<b>Francia</b>	fRO
<b>India</b>	inRO
<b>Indonesia</b>	idRO
<b>Internacional</b>	iRO
<b>Corea</b>	kRO
<b>Malasia</b>	mRO
<b>Filipinas</b>	pRO
<b>Rusia</b>	rRO
<b>Taiwán</b>	twRO
<b>Tailandia</b>	tRO
<b>Vietnam</b>	vRO

Tabla 1: Servidores oficiales de RO

La mecánica en todos los servidores es la misma, pero difieren en las actualizaciones (llegan antes a unos que a otros, y hay algunos que omiten ciertas novedades) y en algunas particularidades del juego. Esto es especialmente notable en el servidor **Valkyrie**, el cual no requiere pagar una cuota para jugar pero tiene muchas diferencias dentro del juego con respecto al servidor de pago<sup>[150]</sup>, que no se enumerarán aquí por ser aspectos muy concretos sin ningún aporte para la memoria, pero que hacen que la experiencia de juego no sea tan buena.



Tampoco el precio es el mismo entre todos los servidores, pero sí es parecido. Como ejemplo se muestran las ofertas de los servidores iRO<sup>[151]</sup> y euRO<sup>[152]</sup> en la *Tabla 2: Precios de los servidores oficiales iRO y euRO*.

	iRO	euRO
<b>30 horas</b>	\$7.99	5.95€
<b>1 mes</b>	\$12	11.95€
<b>3 meses</b>	\$32	29.95€
<b>6 meses</b>	\$57	57.95€

Tabla 2: Precios de los servidores oficiales iRO y euRO

Del mismo modo, pueden encontrarse distintos precios para la compra de ítems y algunos servicios especiales, pero siguen siendo parecidos entre los distintos servidores. Es diferente el caso de Valkyrie, que al no ser de pago estos ítems y servicios son un poco más caros que en los demás servidores.

### 6.4.2 Privados

Cualquier servidor de Ragnarok no oficial es un servidor privado. Existe una gran cantidad de servidores privados por todo el mundo, aunque muchos de ellos sólo llegan a mantenerse abiertos unos pocos meses a lo sumo. Esto es debido a que es relativamente sencillo ofrecer un servidor básico, pero es difícil gestionarlo correctamente u ofrecer algo lo suficientemente interesante como para que los jugadores que lo prueban decidan quedarse y recomendarlo a otros.

La principal ventaja de los servidores privados es que son gratuitos. Normalmente se ofrece la posibilidad de donar para ayudar a los administradores con los gastos mensuales, y en muchos casos se ofrecen ítems y otros servicios o ayudas a cambio de dinero real, igual que en los servidores oficiales. Dependiendo del servidor este intercambio puede dar más o menos ventajas al pagador con respecto a usuarios que no han pagado, desde ceñirse a entregar objetos que en el servidor oficial se consiguen pagando, hasta la posibilidad de obtener ítems personalizados extremadamente poderosos.

Los servidores privados generalmente se publicitan con anuncios en foros (ya sean escritos o pagando por un banner<sup>[70]</sup>) y mediante **tops**<sup>[153][154][155][156]</sup>, páginas web donde pueden darse de alta de forma gratuita y que permiten a los jugadores votar por el servidor. Aquellos con más votos aparecerán los primeros en la página y será más fácil que alguien que busca un servidor lo vea y entre a probarlo. Normalmente los tops poseen algunos filtros para mostrar sólo los servidores con ciertas características, generalmente por rates, pero también los hay que permiten filtrarlos por localización, idioma, etc.

Se puede observar parte de un top en la *Imagen 10: El top de Devstorm Network*.



### TOP Ragnarok Online

Servidores de Ragnarok Online hispanos con todo tipo de rates.

Rank	Título - Descripción	Este Mes
1 (stats)	 <p><b>ShadowRO</b> - Nuevos Rates 6x/6x/7x (cards normales x13, equipo minimo 0.13%)   Oferta de transferencia para Clanes   Gran actualización  Nuevo emulador mejorado  Servidor con más 6 años de experiencia  Staff dedicado   Eventos frecuentes   Novedades únicas            Categoría: Low Rates · Valoración: 5/5 ★★★★★</p>	2281
2 (stats)	 <p><b>Bidai RO - 7/7/7</b> - Quests repetibles   WOES y Novice WOES   Hidden Slot Enchant y Battlegrounds   Exclusivo Pvp 99   Hats Custom   Actualizaciones mensuales   Conquistas, Instancias y Endless Tower   Trivial   Chat Global   GMs dedicados   Más en la web            Categoría: Low Rates · Valoración: 5/5 ★★★★★</p>	1052
3 (stats)	 <p><b>Delate plz RO!</b> - Delate plz RO!            Categoría: Mid Rates · Valoración: 3/5 ★★★</p>	750

Imagen 10: El top de Devstorm Network

Los tops pueden resultar engañosos, por una parte porque en muchos servidores se vota de forma ilegal utilizando proxys<sup>[71]</sup> (no es raro que un servidor recién estrenado suba posiciones en un top de forma escandalosa, pudiendo incluso llegar al primer puesto sólo para desaparecer al cabo de unos meses como si nunca hubiera existido), y por otra porque por defecto se ordenan con los rates mezclados, por lo que los servidores con rates más altos tienden a salir en las primeras posiciones, ya que son los preferidos por la mayoría de los usuarios y por ello reciben más votos. Aun así siguen siendo la herramienta más utilizada para buscar servidores y no ofrecen malos resultados, sobre todo si se utilizan correctamente los filtros para acotar la búsqueda.

## **7 Estructura del servidor BidaiRO**

## 7.1 Portal web

El portal de BidaiRO puede dividirse en tres partes: una con información general sobre el servidor y formularios para realizar ciertas gestiones, el foro y el panel de control. La web está realizada principalmente en **HTML**<sup>[72]</sup>, **PHP 5.2**<sup>[73]</sup> y **Flash**, todo corriendo bajo un servidor **Apache2**<sup>[74]</sup>. Antiguamente también se utilizó **Python**<sup>[75]</sup> para implementar algunas funcionalidades, pero más tarde dejaron de ser necesarias y ya no se utiliza.

El servidor cuenta con el dominio<sup>[76]</sup> **bidairo.com**, registrado mediante **DomainIt**<sup>[158]</sup>, y en la *Tabla 3: Subdominios web de BidaiRO* pueden observarse todos los subdominios accesibles desde cualquier navegador.

URL	Descripción
<b>www.bidairo.com</b>	Página principal
<b>foro.bidairo.com</b>	Foro
<b>cp.bidairo.com</b>	Panel de control
<b>pma.bidairo.com</b>	Administración del motor de bases de datos.

Tabla 3: Subdominios web de BidaiRO

Los tres primeros son de uso público, mientras que el último enlaza a la herramienta **phpMyAdmin**<sup>[77]</sup> para la administración de la base de datos a través de la web, aunque no se suele utilizar.

La estructura de la web es la misma en todos los casos, salvo en el foro y el panel de control. Consiste en una cabecera en Flash con una imagen distinta según la sección de la web que se esté visitando, con un menú donde se puede elegir la sección que se quiere visitar. Cabe destacar la imagen de la página principal, que fue realizada en mayo de 2010 por un usuario que ganó un concurso organizado para sustituir la que había anteriormente. Con estas iniciativas se pretende que los usuarios se sientan una parte activa del servidor.

Seguidamente se muestra una indicación de en qué parte del portal se encuentra el usuario, unos enlaces a los cuatro tops a los que está suscrito el servidor, y otro menú para poder visitar la secciones más importantes de la web.

A la derecha, siguiendo una alineación vertical, se muestra si el servidor está funcionando o no junto al número de usuarios jugando actualmente, información general de cualquier servidor que suele buscar un jugador, así como la hora actual, y otro menú más completo con todas las secciones disponibles en la web, así como sus subsecciones.

En el centro de la web es donde el contenido varía dependiendo de la sección.

La web se aloja en la misma máquina que el resto del servidor. Esto quiere decir que si alguna vez la máquina no está disponible por el motivo que sea (problema de hardware, de la Red, reinicio, etc.) el portal tampoco será accesible. Muchos servidores alojan su web en una máquina aparte para evitar esto, de forma que sólo quedan incomunicados si fallan los dos servicios, cosa poco probable. En BidaiRO no se utilizó este sistema principalmente por cuestiones económicas, ya que es más caro contratar dos servidores que uno. Para cubrir este problema se ha creado una **lista de correo electrónico**<sup>[78]</sup> y se invita a todo el jugador que se registre a unirse a ella, de forma que si ocurre algo basta con comunicarlo por correo. Se ha tenido que hacer uso de esta lista en contadas ocasiones pero con buenos resultados, ya que aunque hay muy poca gente suscrita, quienes reciben la información se encargan de informar a los demás. Para la lista de correo se ha utilizado el servicio gratuito de **FreeLists**<sup>[157]</sup>.

A continuación se muestra la *Imagen 11: Página principal de BidaiRO*, la página que se carga al introducir la dirección **www.bidairo.com** en el navegador.



Web recomendada para navegadores **IE Explorer > 7.0, Netscape > 7.2, Mozilla Firefox u Opera + Plugin FlashPlayer**

# BidaiRO


VIAJA CON NOSOTROS

PRINCIPAL | INFO SERVER | DESCARGAS | JUGAR AQUI | FORO

Principal | Noticias | Actualizaciones | Contactanos

Bienvenido a BidaiRO, tu servidor español de Ragnarok Online con rates 7/7/7, ¡adéntrate!

Te encuentras en: **Principal**




[Información del Servidor](#)
[Panel de Control](#)
[Normas](#)

[Creación de Cuentas](#)
[Descargas](#)
[Foro](#)

## NOTICIAS

**Noticias Actualizaciones Anuncios InGame Contáctanos**


### Mundial de fútbol y WoE



Debido a que el miércoles 7 son las semifinales del Mundial y juega España y el domingo 11 es la final, hemos decidido suspender estas dos WoE ya que prevemos una baja asistencia. La situación del castillo se congelará, de forma que no se puedan obtener cofres ni invertir a partir del día 7, y en su lugar se activará un PvP 99 para los que quieran asistir.

Leer más Enviado por **GM A** el 05/07/2010 a las 19:37

### Boda entre Mariko y Nagato



Mañana martes 6 de julio a las 21:00 (hora española) se celebrará en Lutie la boda entre Mariko y Nagato.

Estáis todos invitados ;)

Leer más Enviado por **Pijus** el 05/07/2010 a las 13:56

### Evento: Endless Tower Weekend

En BidaiRO queremos que disfrutéis y por

## SERVER STATUS

Login Server: **on**  
 Char Server: **on**  
 Map Server: **on**  
 Guardian: **on**  
 Users Online: **73**  
[Panel de Control](#)

## INFORMACION

Rates: **7/7/7**  
 WOEs:  
 \* Miércoles: 22:00-23:00  
 \* Domingos: 22:00-23:00  
 Trivial:  
 \* Martes: 22:00-23:00  
 TimeZone: **GMT+1**  
 ServerTime: 21:57:29  
 LocalTime: 21:57:29

## SECCIONES

- Principal**
  - Noticias
  - Actualizaciones
  - Contactanos
- Info Server**
  - Información
  - Descargas

Imagen 11: Página principal de BidaiRO

Como se puede ver en la Imagen 11, la única parte del portal que está hecha en Flash es la cabecera, el resto es HTML/PHP. Esto repercute en un buen posicionamiento en **Google**<sup>[159]</sup>, ya que sólo puede indexar texto y si se hubiera hecho la mayoría de la web en Flash se saldría perdiendo en ese aspecto. Gracias a esto y a la popularidad del servidor, es posible encontrarlo el primero en el ranking o en la primera página, dependiendo de lo que se busque, como se puede observar en la *Imagen 12: BidaiRO en la primera posición del Google*.



**Imagen 12: BidaiRO en la primera posición del Google**

### 7.1.1 Mapa web

A continuación se muestra la *Imagen 13: Mapa de la web de BidaiRO*, donde se puede apreciar de forma sencilla toda la estructura de la web, que se describirá más adelante con detalle.



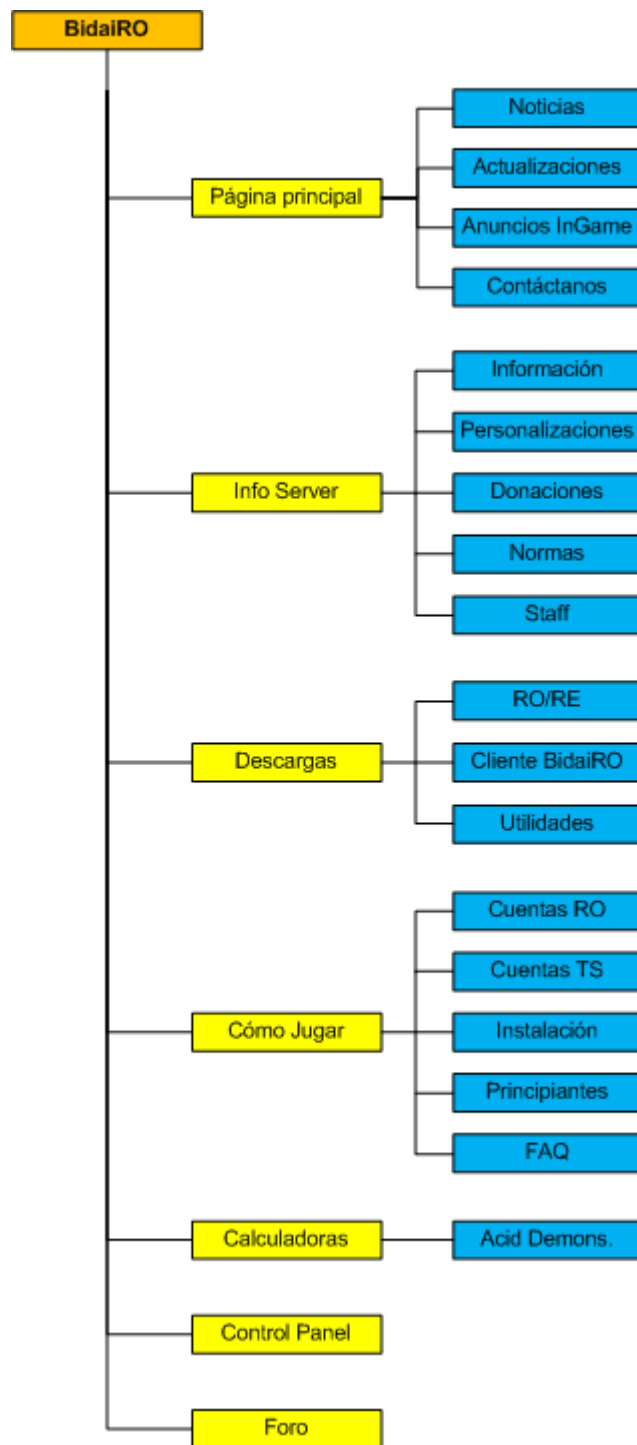


Imagen 13: Mapa de la web de BidaiRO

### 7.1.2 Sección *Principal*

Aquí se muestran los anuncios importantes del servidor de forma dinámica, y se proporciona un enlace a un formulario de contactos para que quien quiera pueda mandar un correo a la administración desde la web. La sección se divide en cuatro puntos:

- **Noticias**

Aquí se muestran anuncios del servidor de carácter general y que afecten al juego, como pueden ser eventos, bodas entre jugadores, concursos y cualquier cosa que no tenga cabida en otras secciones más específicas.

- **Actualizaciones**

Esta sección se utiliza exclusivamente para anunciar las actualizaciones que se realizan en el servidor, enumerando las novedades y los cambios realizados. Si algún cambio supone una novedad importante, es posible publicarlo también en la sección de *Noticias* y referenciarlo desde esta sección.

- **Anuncios InGame**

Este término se utiliza para describir algo que sucede dentro del propio juego. La sección se llama así porque el contenido del anuncio también aparecerá en una ventana la primera vez que un jugador entre al juego. Normalmente se utiliza para avisar de cuándo será la próxima actualización o para mostrar anuncios importantes, ya que no todos los usuarios visitan el foro. El texto aquí publicado debe tener como máximo 255 caracteres, por lo que no es raro publicar un resumen de algo de importancia y pedir a los usuarios que visiten alguna sección del foro donde se explica con más detalle.

- **Contáctanos**

Esta sección consiste en un formulario estático donde los usuarios pueden comunicarse con la administración de BidaiRO seleccionando la categoría a la que pertenece su consulta. Los formularios enviados de esta forma se enviarán a la cuenta de **Gmail**<sup>[160]</sup> del servidor,

**bidairo@gmail.com**, etiquetados según la categoría elegida para una mejor organización de los emails.

La decisión de utilizar una cuenta de Gmail en vez de crear una dirección de correo utilizando el dominio del servidor se tomó porque muchos servidores de correo catalogaban de spam<sup>[79]</sup> los emails enviados desde bidairo.com. Para evitar gestiones con dichos servidores se utilizó Gmail, que además ofrece muy buenas herramientas para organizar los correos.

### 7.1.3 Sección *Info Server*

Esta sección está compuesta por varias subsecciones donde se da información de diversa índole acerca del servidor, sus características y la administración. Está dividida en cinco puntos:

- **Información**

En esta sección se muestra información sobre la máquina en la que corre el servidor y sobre las características del servidor en sí que todo jugador busca antes de empezar a jugar, como los rates, horario de WoEs y características especiales del servidor. Más adelante la lista continúa con un poco más de detalle.

- **Personalizaciones**

Aquí se muestra una lista de las características que sólo tiene BidaiRO o que no existen por defecto en un servidor oficial. Hay muchas cosas que se han estandarizado en los servidores privados y por tanto no son exclusivas de este, pero se detallan también aquí. Muchas de las descripciones hacen referencia a publicaciones del foro, donde se describe la personalización con más detalle.

- **Donaciones**

Existe la posibilidad de donar para ayudar a pagar los gastos del servidor, ya sea mediante **transferencia bancaria** a una cuenta de **Cajamadrid** abierta específicamente para gestionar los pagos e ingresos

del servidor, o mediante **PayPal**. Al donante se le entregarán un número de **Bidai Coins** dependiendo de la cantidad donada, que consisten en unas monedas virtuales que pueden ser utilizadas dentro del juego para obtener ciertas ventajas no desequilibrantes, como puede ser nuevos colores para la ropa, el pelo, cambios de nombres de personajes, etc. En esta sección se presentan todos los datos y pasos necesarios para realizar una donación y las ventajas que se pueden obtener.

- **Normas**

Es necesario establecer unas normas para mejorar la convivencia y aclarar qué cosas pueden hacerse y cuáles no en caso de que pudieran existir dudas. En este apartado se enumeran las normas que se deben respetar al utilizar cualquier servicio de los ofrecidos por el servidor.

- **Staff**

En esta sección se muestran las personas encargadas de las tareas administrativas del servidor, así como los roles generales que desempeña cada uno. También se ofrece un enlace al formulario de contactos para que el usuario pueda acceder fácilmente si necesita realizar cualquier tipo de consulta.

#### **7.1.4 Sección Descargas**

En esta sección se proporcionan enlaces a todo el software necesario para jugar, y a algunas utilidades. Está compuesto por tres subsecciones:

- **RO/RE**

En esta sección se puede descargar la primera parte necesaria para jugar, correspondiente a los clientes Ragnarok y Renewal. Se explicará con detalle más adelante, en el capítulo *Cliente de Gravity*. Los ficheros son grandes y están alojados en servidores externos gratuitos para no consumir ancho de banda extra.

- **Ciente BidaiRO**

En esta sección se puede descargar la otra parte necesaria para jugar, correspondiente al cliente de BidaiRO. Se explicará con detalle más adelante, en el capítulo *Ciente BidaiRO*. Los ficheros son grandes y están alojados en servidores externos gratuitos para no consumir ancho de banda extra.

- **Utilidades**

Aquí se ofrecen enlaces a algunas utilidades comunes tanto fuera como dentro del mundo del Ragnarok, como pueden ser gestores de descarga, descompresores, programas para grabar vídeos y otros más específicos para el juego.

### 7.1.5      **Sección *Cómo Jugar***

Esta sección es importante porque desde aquí, además de varias guías básicas para quienes no conozcan el funcionamiento del juego, se encuentran los formularios de creación de cuentas. Consiste en cinco subsecciones:

- **Cuentas RO**

Desde aquí es posible crear la cuenta del juego, que deberá ir asociada a una cuenta del foro previamente creada, de forma que se pueda llevar un seguimiento de qué publica en el foro cada jugador. También existe un formulario para solicitar permiso para poder jugar con más de una cuenta desde la misma IP, ya que por defecto el acceso está restringido a una para que los usuarios no jueguen con dos cuentas al mismo tiempo.

- **Cuentas TS**

Al igual que en el apartado anterior, desde aquí es posible crear una cuenta para el servidor **TeamSpeak**<sup>[80][161]</sup>, que deberá ir asociada a la cuenta del juego. Obviamente es imprescindible crear al menos la cuenta de juego para poder jugar (y por tanto la del foro), pero la de TeamSpeak no es obligatoria, aunque sí recomendable para poder aprovechar las ventajas de la comunicación por voz.

- **Instalación**

En esta sección se describen los pasos necesarios para instalar el cliente. La instalación lleva al menos dos pasos y no es complicada, pero los instaladores de Ragnarok y Renewal están en coreano, por lo que nunca está de más una guía. Además también se explica cómo actualizar el juego una vez instalado, algo que muchos jugadores no hacen.

- **Principiantes**

Desde aquí se hace una pequeña guía del funcionamiento del cliente y del juego, no tan extensa como el contenido del capítulo *Ragnarok Online* de este documento, lo suficientemente explicativo como para que un novato pueda dar sus primeros pasos. Se describen las clases principales, los atributos y la forma de crear un personaje, así como los distintos elementos de la ventana de juego.

- **FAQ**

Acónimo de Frequent Asked Questions (*preguntas más frecuentes*), en esta sección se plantean algunas preguntas con sus correspondientes respuestas sobre el juego y el servidor.

### **7.1.6 Sección de calculadoras**

Esta sección se creó a raíz de la colaboración de un usuario para añadir calculadoras de diferentes aspectos del juego. Dado que es un sistema con un alto componente numérico y probabilístico, es muy común utilizar calculadoras para obtener ciertos datos, como por ejemplo la probabilidad de crear una poción con éxito o la cantidad de vida que se puede curar con un hechizo de curación, todo ello dependiendo de varias variables como el nivel, el equipo, habilidades, etc., que se le suministran a la calculadora.

Por el momento sólo se cuenta con la subsección correspondiente a la calculadora que remitió este usuario ya que nadie envió ninguna más, pero la sección está ahí para ampliarla si alguien quisiera colaborar.:

- **Acid Demons.**

Acid Demonstration es la habilidad principal de la clase *Creator*, y es de las más letales para personajes o monstruos con mucha vitalidad, por ello es muy útil una calculadora. Esta está orientada a WoE y es posible seleccionar todas las opciones que pueden modificar el daño realizado o recibido, tanto para un jugador como para un monstruo, dependiendo de los atributos y el equipo utilizado.

### 7.1.7 Panel de control

El panel de control es una sección imprescindible en un servidor, ya que permite a los usuarios la posibilidad de gestionar ciertos aspectos de sus cuentas y obtener información del juego. Este tipo de información suele consistir en rankings donde se puede consultar quién es el personaje con más nivel, qué clanes han conseguido más castillos, etc.

El panel de control utilizado es **Ceres**<sup>[162]</sup>, uno de los más famosos y utilizados, aunque hoy en día en los servidores nuevos está cayendo en desuso por otros con mejores prestaciones (al fin y al cabo hace años que Ceres no se actualiza). A la base de Ceres se le han añadido varias personalizaciones, sobre todo una gran variedad de rankings para los usuarios, así como una lista mejorada de jugadores conectados, y también algunas opciones de gestión administrativas. Ceres obtiene las credenciales de cada jugador directamente de la base de datos del juego, por lo que quien tenga una cuenta de administrador podrá acceder a todas las opciones pero el resto de jugadores no las verán.

También se le ha añadido la posibilidad de votar en los tops. Aprovechando que el panel de control está asociado a la cuenta de cada jugador, cada votación realizada desde ahí proporciona a esa cuenta un **punto de votación**<sup>[81]</sup> o **vote point**, que se puede cambiar dentro del juego por algunas ayudas temporales y muy limitadas, de forma que se incentiven las votaciones.

### 7.1.8 Foro

Aquí es donde se anuncian todas las noticias del servidor, y el canal de comunicación preferido para que los usuarios contacten con la administración. Además de eso es donde ellos mismos pueden debatir los temas que les interesen, tanto si son referentes al juego como si no, siempre que lo hagan en las secciones adecuadas.

El foro es un **phpBB**<sup>[82][163]</sup> con varias modificaciones. La más evidente de cara al usuario es el **Bugtracker**<sup>[83]</sup>, mostrado en la *Imagen 14: Bugtracker*, consistente en un subforo modificado para poder llevar un seguimiento de los bugs<sup>[84]</sup> reportados por los usuarios de forma clara y ordenada. Estos pueden enviar los bugs que encuentren directamente desde el juego o abriendo un nuevo tema en el propio Bugtracker. En el caso de que lo envíen desde el juego, llegará a un Bugtracker interno sólo para administradores, y una vez leído se moverá al Bugtracker público.

Revisando	Corrigiendo		Corregido	
En espera	Bug leve		No es Bug	
Revisando	Bug grave		Corregido	
Sin Solución	Reportado en eA		Información	

New Topic









Estado	Temas	Gravedad	Autor	Último Mensaje	
	-	Nota (PostIt): Bugtracker	0	Luffy	Vie Dic 12, 2008 1:50 am Luffy ➡
	No es un bug	Imposibilidad de renacer	0	Mustaine	Sab Jul 10, 2010 5:31 pm Mustaine ➡
	No es un bug	Y los antelope horns del tool dealer de niff??	0	hekki	Jue Jul 08, 2010 7:26 pm hekki ➡
	No es un bug	Nuevos items que no salen?	0	hekki	Jue Jul 08, 2010 7:11 pm hekki ➡
	Corregido	Bug quest repetible de anolian skins	1	hekki	Jue Jul 08, 2010 7:02 pm GM A ➡
	No es un bug	Quest Curse of Gaebolg Family Opcion 2 Father Bamph	0	Bugtracker	Mie Jul 07, 2010 11:27 pm Laerians_Nycht ➡
	Corregido	PvP 99: Stalker.	1	Bugtracker	Dom Jul 04, 2010 2:02 pm GM A ➡
	Sin Solución	Suelo	1	Bugtracker	Dom Jun 20, 2010 6:01 pm GM A ➡

Imagen 14: Bugtracker



En la *Imagen 15: Reporte de un bug en el Bugtracker* se puede ver el resultado de un bug reportado por un usuario.

Índice del Foro BidaiRO > Bugtracker > PvP 99: Stalker.

Su última visita fue el Mar Jul 27, 2010 2:21 pm  
Fecha y hora actual: Mar Jul 27, 2010 7:27 pm

Ver mensajes desde su última visita  
Ver sus mensajes  
Ver mensajes sin respuestas

Estado del Bug

Estado	Corregido	Reportado por	Blood Lemon
Mapa:	prontera,133,205	Char ID:	153237
Gravedad	1	Revisión	0

Cambia el estado del bug
Cambia la gravedad del bug

New Topic
Post Reply

Ver tema anterior :: Ver siguiente tema

Autor	Mensaje
<b>Bugtracker</b> Soluciona Problemas  Registrado: 14 Dic 2008 Mensajes: 77 Volver arriba	Publicado: Lun Jun 28, 2010 10:07 pm quote edit x ip Si se copia alguna skill siendo stalker, y en una edición siguiente se cambia de job, la skill copiada se mantiene , profile pm email
<b>GM A</b> A Mí Sí Me Gusta Volar  Registrado: 09 Dic 2008 Mensajes: 3414 Volver arriba	Publicado: Dom Jul 04, 2010 2:02 pm quote edit x ip Ya debería estar arreglado.  profile pm email

Imagen 15: Reporte de un bug en el Bugtracker

También cuenta con un filtro anti-spam de creación propia muy efectivo. Antes de su implementación era muy común recibir varios mensajes de spam al día, pero después, tras unos pequeños ajustes dejaron de llegar. El filtro consiste simplemente en validar la primera publicación de cada usuario contra una lista de palabras, y si alguna

coincide se le deshabilita la cuenta del foro y se borra la publicación. El chequeo sólo se realiza una vez, de forma que apenas se han recibido falsos positivos<sup>[85]</sup>.

El foro también informa de cuándo ha llegado un correo a la cuenta de Gmail del servidor. Aprovechando que los correos se reciben a través del formulario de contactos de la web, se ha redirigido la misma información que se envía a Gmail a un subforo específico para ello. Esto es útil porque permite discutir el tema entre todos los administradores por medio del foro para cada correo recibido de forma individual, sin “ensuciar” el original. Para contestar al correo sigue siendo necesario hacerlo a través de Gmail.

La versión de phpBB utilizada es la 2, la última que había en el momento de abrir el servidor. Actualmente está disponible la versión 3, pero se ha mantenido la anterior porque se considera que con las funcionalidades que ofrece se cubren todas las necesidades, y migrar de versión sería una tarea costosa, sobre todo por las modificaciones que se le han añadido.

En el *Anexo B: Estructura del foro* se puede consultar la estructura del foro.

### **7.1.9 Mapa de interacción**

A continuación se muestra la *Imagen 16: Estructura del portal web*, un diagrama para facilitar la comprensión de la interacción entre los distintos componentes del portal web.

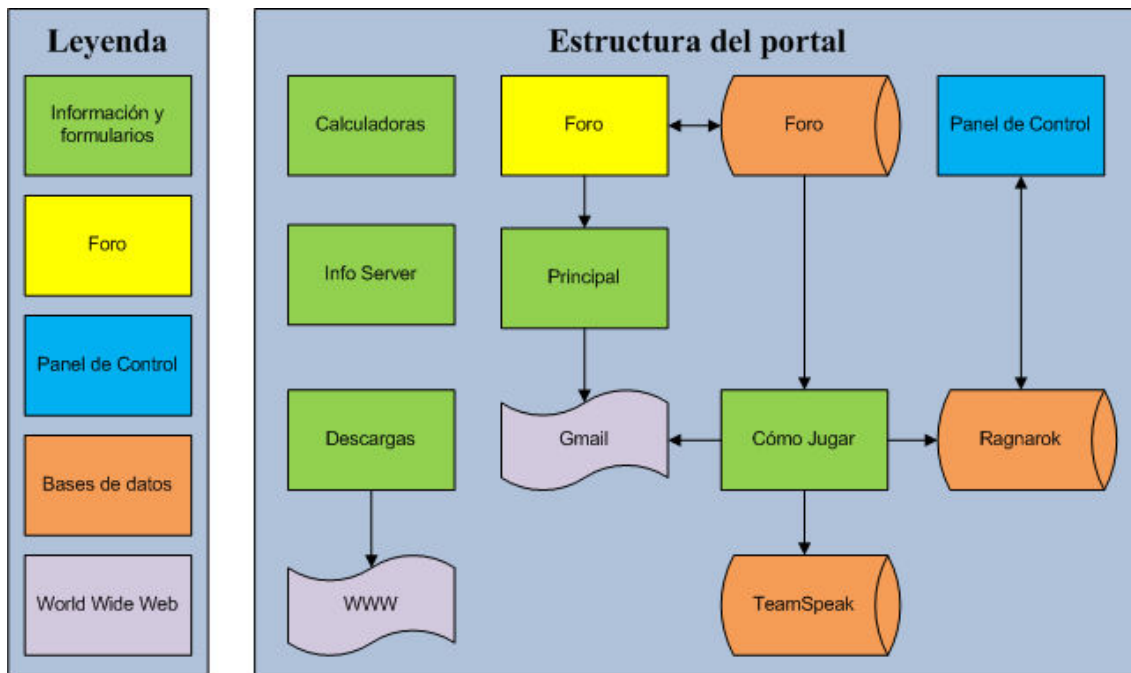


Imagen 16: Estructura del portal web

Como se comentó anteriormente, la página principal muestra los contenidos del foro correspondientes a las secciones de **Noticias**, **Actualizaciones** y **Anuncios InGame**, y además da la posibilidad de comunicarse con la administración del servidor a través de un formulario que envía un correo a la cuenta de **Gmail**.

Desde la sección *Cómo Jugar*, donde se encuentran los formularios de creación de cuentas y petición de permisos para poder jugar con más de una cuenta desde la misma IP, se observa una relación entre las bases de datos necesarias para gestionar estos formularios, así como la posibilidad de enviar un correo a la administración para solicitar los permisos.

Por último la sección de *Descargas* referencia a enlaces externos para obtener todo el software ofrecido por el servidor. El foro y el Panel de Control utilizan la información de sus bases de datos correspondientes, y las secciones de *Calculadoras* e *Info Server* no tienen interacción con otros componentes.

Seguidamente, en la *Imagen 17: Diagrama de secuencia de creación de cuenta*, se muestran los pasos necesarios para que un nuevo usuario comience al jugar. En ellos se aprecia la interacción entre algunos componentes del portal, el emulador y la base de datos. Estos últimos se describen con detalle a continuación.

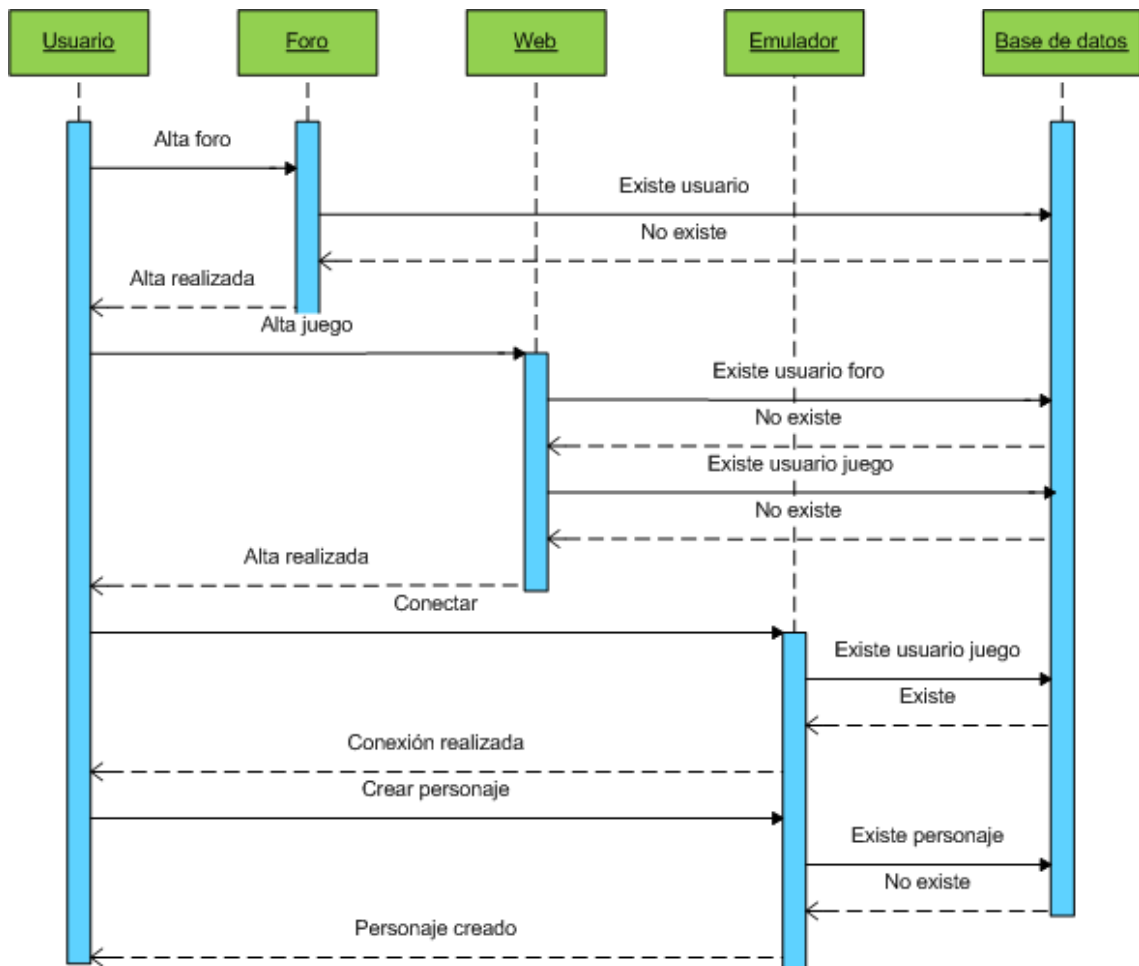


Imagen 17: Diagrama de secuencia de creación de cuenta

## 7.2 Servidor eAthena

El Ragnarok Online utiliza una arquitectura de tipo **cliente-servidor**, por lo que para poder jugar es indispensable disponer de un cliente en la máquina del usuario, y de un servidor donde conectarlo. **eAthena**<sup>[86][164]</sup> es el nombre de ese servidor, más concretamente del equipo que lo gestiona y actualiza. eAthena es un emulador de código libre escrito en C<sup>[91]</sup> del servidor oficial de Gravity, llamado **Aegis**<sup>[87][165]</sup>.

eAthena no se creó de cero, sino que tomó Aegis como base cuando este se empezó a filtrar. Hay que destacar que mientras que la utilización de Aegis es ilegal, ya que es un software propietario de Gravity, la de eAthena es completamente legal al ser software libre. Aun así, existen muchos servidores privados que utilizan Aegis, pero la gran mayoría corren bajo eAthena. Ha habido algún proyecto pequeño de emulador pero ninguno ha cosechado éxito, por lo que se puede afirmar que sólo existen dos emuladores de Ragnarok en circulación.

Pese a haber sólo dos emuladores, es habitual encontrar forks<sup>[88]</sup> de eAthena, que en muchas ocasiones se acaban uniendo al desarrollo principal cuando han terminado de implementar su desarrollo paralelo. Actualmente se está desarrollando el **3CeAM**<sup>[89][166]</sup>, un fork que está implementando las terceras clases, ya que en eAthena están trabajando en otras cosas y no parece que vayan a avanzar demasiado en eso a corto plazo. Es de esperar que cuando 3CeAM finalice su trabajo se incluirá en eAthena, como se ha hecho en otras ocasiones.

eAthena utiliza **Subversion** (SVN) como sistema de control de versiones, y utiliza dos ramas<sup>[90]</sup> principales, **stable** y **trunk**. La diferencia entre ambas ramas reside en que la primera ofrece más garantías de estabilidad que la segunda, pero esta última se actualiza con mayor frecuencia. Una vez que el equipo de eAthena se asegurara de que las novedades implementadas en la rama trunk fueran estables, las pasaban a la rama stable.

Actualmente la rama stable ha quedado en desuso y es altamente desaconsejable utilizarla por estar muy poco actualizada (el último cambio data del 23 de abril de 2010,

y contiene sólo una mínima parte de lo que ofrecía la trunk por aquel entonces). En cambio la rama trunk ha ganado en calidad y es muy raro encontrar errores en ella. En BidaiRO se empezó usando la rama stable, pero desde julio de 2009 se empezó a utilizar la trunk. El emulador del servidor se actualiza con las novedades de eAthena al menos una vez a finales de mes, y a veces, si hay actualizaciones lo suficientemente interesantes, a mediados. También se pueden realizar actualizaciones fuera de calendario si son críticas (los cambios de eAthena se revisan a diario, por lo que el tiempo de respuesta es muy alto).

Las actualizaciones se realizan mediante un script para Linux de diseño propio. Dado que en el servidor de BidaiRO se utiliza un SVN distinto del de eAthena para poder desarrollar en paralelo, es necesario unir la información de ambos para poder actualizar correctamente el código de eAthena sobre el de BidaiRO. Gracias al script, las actualizaciones se realizan de forma casi automática y se ahorra mucho tiempo y esfuerzo. Los únicos casos en los que hay que intervenir manualmente es cuando el SVN encuentra algún conflicto que no puede resolver durante la actualización. En el *Anexo C: Script de actualización de eAthena* se muestra el código del script.

eAthena viene en dos modalidades, una donde se almacenan los datos en ficheros de texto plano y otra donde se almacenan en bases de datos. Normalmente se utiliza la primera para servidores caseros, ya que es más sencillo hacerlos funcionar. En BidaiRO se utiliza la versión para base de datos, debido a las múltiples ventajas que ofrece frente a los ficheros de texto plano. Aun así, mucha información se sigue obteniendo de ficheros, como todo lo referente a la configuración del emulador y a algunos datos que sólo se cargan en memoria cuando se arranca este. Probablemente los desarrolladores de eAthena comenzaran utilizando sólo ficheros de texto plano, migrando con el tiempo al sistema de base de datos los puntos más críticos, pero dejando de lado el tratamiento de la información que sólo se lee una vez para guardarla en memoria por no afectar al rendimiento del emulador.

El emulador original está dividido en tres servidores, cuatro en el caso de BidaiRO. Estos son el **login server**, **char server** y **map server** (*servidor de ingreso, servidor de personaje y servidor de mapa*). Cada uno tiene una funcionalidad concreta y aunque interactúan entre ellos, son independientes entre sí, de forma que es posible

reiniciarlos de forma individual sin que el resto se vean afectados (cosa poco común pero que a veces se ha tenido que hacer para arreglar algún problema sin causar un gran impacto al resto del servidor). Tanto es así que cada servidor tiene su propio fichero de configuración. Además de los tres servidores originales, en BidaiRO existe el **anticheat server** (*servidor antitrampas*), igualmente independiente de los anteriores.

En la *Imagen 18: Interacción básica del servidor eAthena en BidaiRO*, se puede observar cómo interactúan entre sí los servidores originales de eAthena, el anticheat y el usuario que quiere conectar al juego.

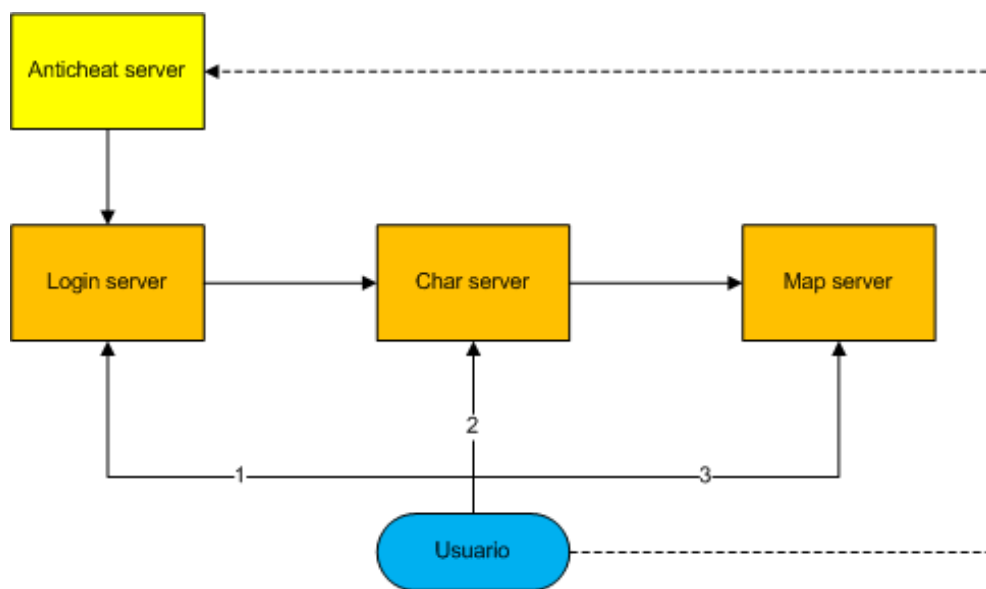


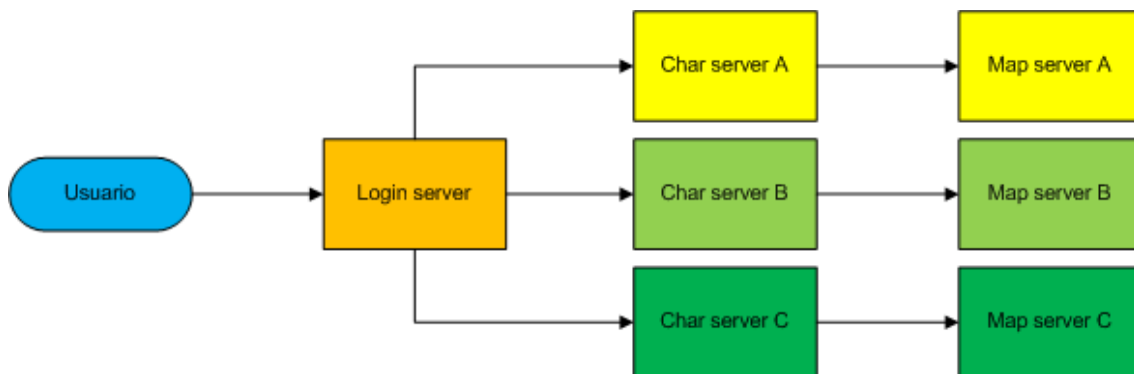
Imagen 18: Interacción básica del servidor eAthena en BidaiRO

### 7.2.1 Login server

Cuando un usuario quiere acceder al servidor, lo primero que hace es conectar al login server desde el cliente. Este servidor se encarga única y exclusivamente de gestionar las peticiones de acceso y los accesos al char server. Cuando un usuario conecta, primero se comprueba si tiene permisos para acceder al resto del juego (la cuenta no debe estar bloqueada, el acceso al servidor no debe estar limitado a administradores, etc.), y si es así se comparan sus credenciales con las de la base de datos (usuario y clave). Si todo es correcto, informa al char server y al cliente y cierra la conexión con este último, que procederá a abrir una nueva con el char server.

Este proceso tiene varias capas de seguridad para evitar que se realicen conexiones al char server sin pasar previamente por el login server, evitar que se realicen dos conexiones simultáneas por parte de la misma cuenta, y otras comprobaciones que podrían causar efectos no deseados en el servidor, tanto para su estabilidad como para el funcionamiento del juego en sí, ya que saltarse estas restricciones es una forma muy común de explotar vulnerabilidades del juego, permitiendo por ejemplo duplicar equipo de forma ilegal.

El login server se encarga también de listar todos los servidores a los que puede conectar el usuario, ya que es posible que un solo login server esté conectado a múltiples char servers con sus correspondientes map servers, como se puede apreciar en la *Imagen 19: Estructura básica de un login server con múltiples char/map servers*. De esta forma se pueden tener varios servidores de Ragnarok con características diferentes, cada uno de ellos compartiendo las mismas cuentas de usuario (información procesada por el login server) pero con diferentes personajes (información procesada por los char servers). Esto permite que el usuario sólo tenga que recordar una cuenta y un password, pero dependiendo del servidor al que acceda tendrá sus propios personajes independientes.



**Imagen 19: Estructura básica de un login server con múltiples char/map servers**

El login server es el servidor que menos recursos consume de los tres, ya que sólo se encarga de tareas sencillas que se ejecutan una única vez cada vez que un jugador quiere acceder al servidor. Si el login server se cae o se deshabilita los jugadores que se encuentren dentro del juego pueden continuar jugando con total normalidad, pero no podrá acceder ningún otro jugador.



### **7.2.2 Char server**

El char server se encarga de mostrar a los jugadores la lista de sus personajes disponibles (siempre asociados a la cuenta con la que hayan accedido), y una vez que se selecciona uno, lo notifica al map server y al cliente y sigue el mismo proceso que el en el caso del login server. Normalmente las vulnerabilidades entre cambios de servidor se realizan aquí, entre el char server y el login server. Hoy en día el equipo de eAthena ha mejorado enormemente la seguridad en este aspecto y se presupone que no es posible vulnerar la seguridad del servidor en este punto. BidaiRO colaboró en este punto a principios de 2009 ayudando a resolver el último fallo detectado de este tipo.

Además de mostrar una lista de personajes disponibles, el char server se encarga de cargar y grabar la información de estos personajes. Cuando un jugador selecciona su personaje para jugar, el char server obtiene toda la información necesaria y la pone a disposición del map server. Igualmente, de forma periódica o en determinadas ocasiones preestablecidas, el char server graba la información de los jugadores en la base de datos. Es por esto que el char server es el segundo en cantidad de recursos consumidos, ya que aunque la selección de personajes no se realice con mucha frecuencia (pero más que las peticiones de acceso del login server, ya que un jugador puede querer cambiar de personaje en cualquier momento), la grabación de los personajes sí se realiza con una frecuencia mayor y maneja un volumen de datos superior.

Si el char server se cae o se deshabilita, los usuarios que ya estén conectados podrán seguir jugando, pero sus progresos no se guardarán. Esto es importante porque es imposible saber si el servidor se ha caído si no es intentando cambiar de personaje, lo cual quiere decir que si el usuario deja de jugar, perderá todo lo que haya hecho a no ser que se restablezca el servicio a tiempo, y aun así al ser un servicio multijugador las probabilidades de que algún usuario se vea afectado son muy altas.

### **7.2.3 Map server**

El map server es el pan de cada día de un servidor de Ragnarok Online. En él se almacena toda la información del juego: los mapas, los monstros, los objetos, los jugadores, los clanes, etc. Cuando el servidor arranca, carga toda esta información en memoria y recibe la información de los jugadores que van entrando del char server.

Toda la lógica y la mecánica del juego está en este servidor, por lo que supera con creces el consumo de recursos de los otros dos.

Al igual que el char server envía información al map server para todos los temas referentes a los personajes, el map server también envía mucha información al char server, de forma que este último pueda actualizar toda la información necesaria de los personajes que están jugando y de otros aspectos globales que puedan afectar al resto de personajes que no están jugando. Esta comunicación, que va a la inversa del flujo que se ha estado viendo a partir de una petición de acceso al servidor por parte de un usuario, no sólo se limita al char server, sino que también puede llegar al login server, por ejemplo para mostrar una alerta de algún administrador en todos los servidores, como se observa en la *Imagen 20: Map server B comunicándose con el resto de map servers*.

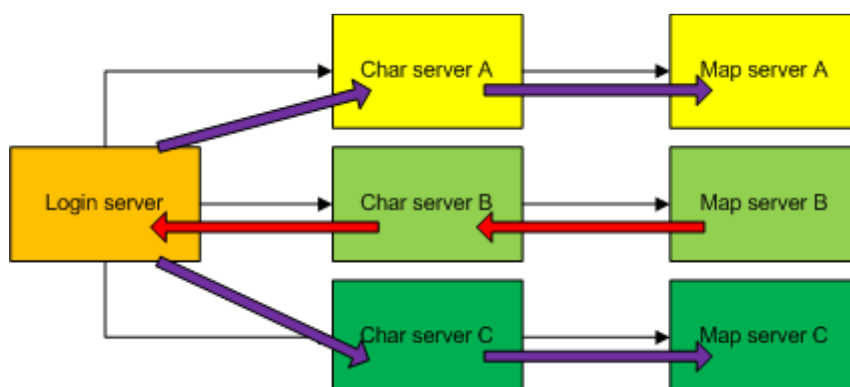


Imagen 20: Map server B comunicándose con el resto de map servers

Como es lógico, si el map server deja de funcionar es imposible jugar. Un usuario que intente conectar llegará hasta el char server, donde podrá seleccionar uno de sus personajes, pero no pasará de ahí. Si el map server se cae, los usuarios que estuvieran conectados perderán lo que hubieran jugado desde la última vez que se grabaron sus personajes, por lo que las pérdidas no serán de más de unos minutos en el peor de los casos (o nada en el mejor de los casos).

En el map server es donde se modifican o se desarrollan la mayoría de las características propias de BidaiRO, ya que es donde se encuentra el código que afecta al sistema de juego en sí. Como ejemplo, en el *Anexo D: Implementación del Trivial Online* se describirá la implementación de un trivial online que se ha añadido al

emulador, y en el *Anexo E: Implementación de eventos* se podrán ver ejemplos de eventos desarrollados en el lenguaje de scripting de eAthena.

#### 7.2.4 Anticheat server

Este servidor, implementado exclusivamente por BidaiRO, es un añadido a los tres servidores de eAthena. El concepto en sí no es nuevo, ya que muchos servidores implementan algún tipo de protección similar para evitar que los jugadores hagan trampas. Obviamente, si eAthena implementara un anticheat propio, no tendría ninguna utilidad puesto que todo el mundo conocería su funcionamiento al ser público y no tardarían en idear algún sistema para evitarlo.

La finalidad principal del servidor anticheat es conocer si el cliente de BidaiRO ha sido modificado o se está utilizando otro distinto. Esto se consigue contrastando varios CRC<sup>[106]</sup> del cliente con la tabla<sup>[107]</sup> *anticheat* de la base de datos, donde se almacenan los valores permitidos. El CRC es un número calculado a partir del contenido del fichero que se quiere analizar, por lo que si este número no coincide con el del fichero original almacenado en la tabla, se puede saber que ha sido modificado. Los componentes cuyo CRC se calculará se describen más adelante a lo largo del capítulo *Cliente*, y en la *Imagen 21: Contenido de la tabla anticheat* se puede ver la información actual almacenada en la tabla.

Tipo	crc	Descripcion
0	1504007909	Cliente GM
0	3932146632	Cliente Normal
1	985517705	grf.dll
2	632577234	GRF custom
3	3839482639	bidai.grf
4	4151010421	BidaiRO.dll
5	1514276957	anticheat.dll V2.4

Imagen 21: Contenido de la tabla anticheat

Para que el servidor anticheat funcione, es necesario una DLL<sup>[92]</sup> que haga de puente entre el cliente y el servidor para poder interpretar los códigos que envía el anticheat y realizar las acciones necesarias. Estas acciones consisten en la identificación

de la ruta donde se encuentran los componentes a analizar y la petición de sus CRC, pero el sistema está pensado para poder obtener más datos. En un futuro se pedirá también un identificador único de Windows que permita identificar de forma unívoca cada copia, para así poder saber en caso necesario a qué usuario pertenece cada cuenta sin depender de la IP, que normalmente es variable. La forma de utilizar esta DLL es inyectándola<sup>[93][167]</sup> en el ejecutable del cliente modificando directamente el binario<sup>[94]</sup>, ya que al no disponer del código fuente del mismo no hay otra manera de incluir las funcionalidades requeridas.

Una vez solventado el problema de la obtención de datos en la parte del cliente, queda ver cómo interactúa el servidor con el resto de eAthena. Para ello es necesario modificar la parte del login server y char server que procesa la autenticación del cliente o del servidor anterior en la cadena, y redirigirlo al servidor anticheat para que haga las comprobaciones correspondientes. Una vez realizadas, se devuelve el flujo al servidor de eAthena y se prosigue con el proceso de autenticación o se deniega el acceso. Implementar esto es mucho más sencillo que en la parte del cliente, ya que se dispone del código fuente de todo el emulador.

En la *Imagen 22: Flujo detallado de la comunicación entre el anticheat y eAthena* se puede apreciar de forma visual los pasos que siguen los servidores para comunicarse entre sí.

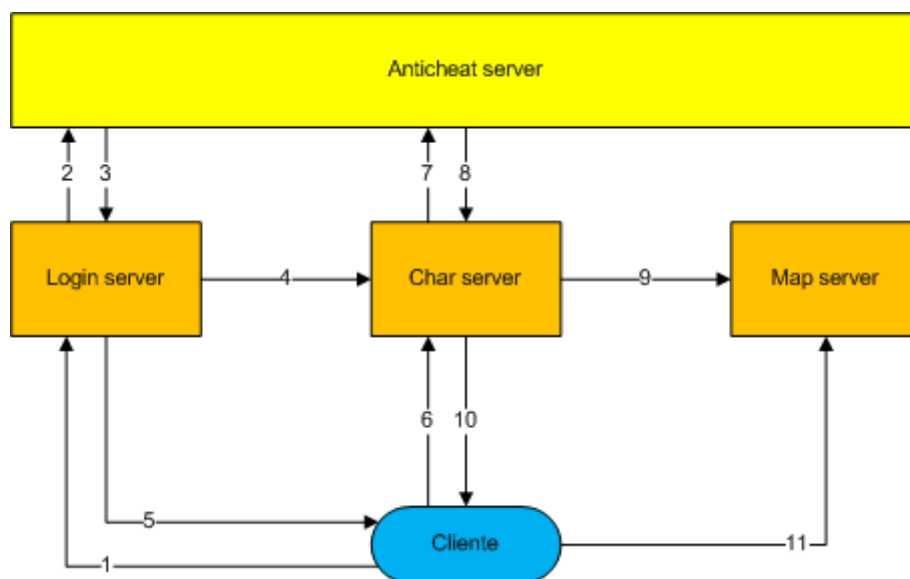


Imagen 22: Flujo detallado de la comunicación entre el anticheat y eAthena

Además de estos chequeos, en el propio map server se ha implementado una comprobación extra a los 20 segundos de haber empezado a jugar, que cerrará la conexión en caso de que algunos de los CRC sean incorrectos o no se hayan analizado. Esta comprobación sirve para reforzar la seguridad, ya que tal y como está diseñado el emulador es posible, aunque raro, que la expulsión del cliente se realice cuando se está pasando de un servidor a otro, en una especie de limbo en donde no tendría efecto.

En la *Imagen 23: Diagrama de secuencia de login*, se muestra con detalle el proceso que sigue un cliente y los servidores para un login válido.

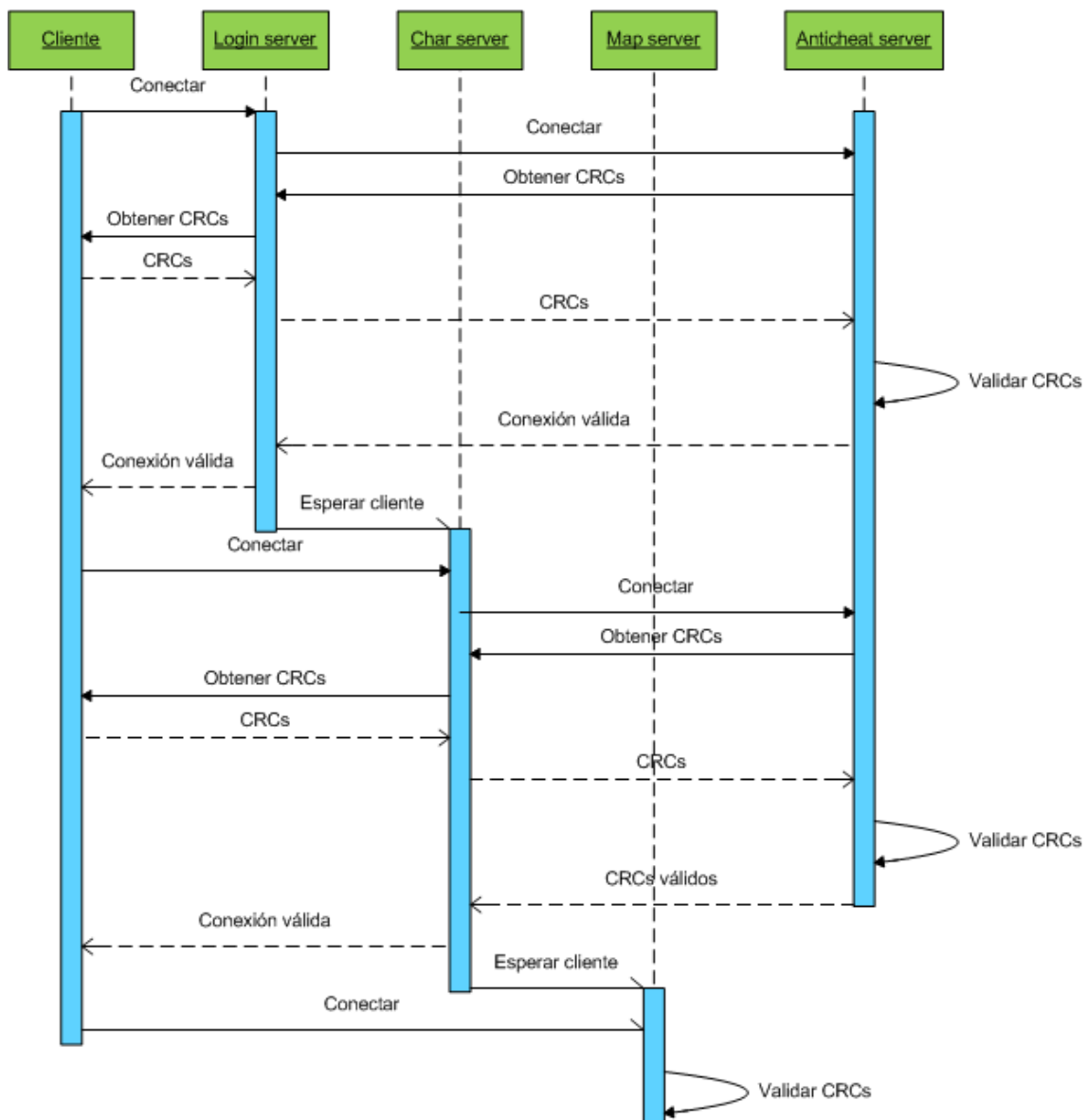


Imagen 23: Diagrama de secuencia de login

## 7.3 *Cliente*

Para poder jugar a Ragnarok Online el usuario debe descargar una serie de software que le permita acceder al servidor. Para ello se utiliza el cliente de Gravity, donde se encuentra el juego en sí, y el cliente de BidaiRO, que permite la conexión con el servidor y contiene distintas personalizaciones con respecto al oficial.

Para jugar, primero debe instalarse el cliente de Gravity y después del cliente de BidaiRO en la misma carpeta donde se haya instalado el anterior.

### 7.3.1 **Cliente de Gravity**

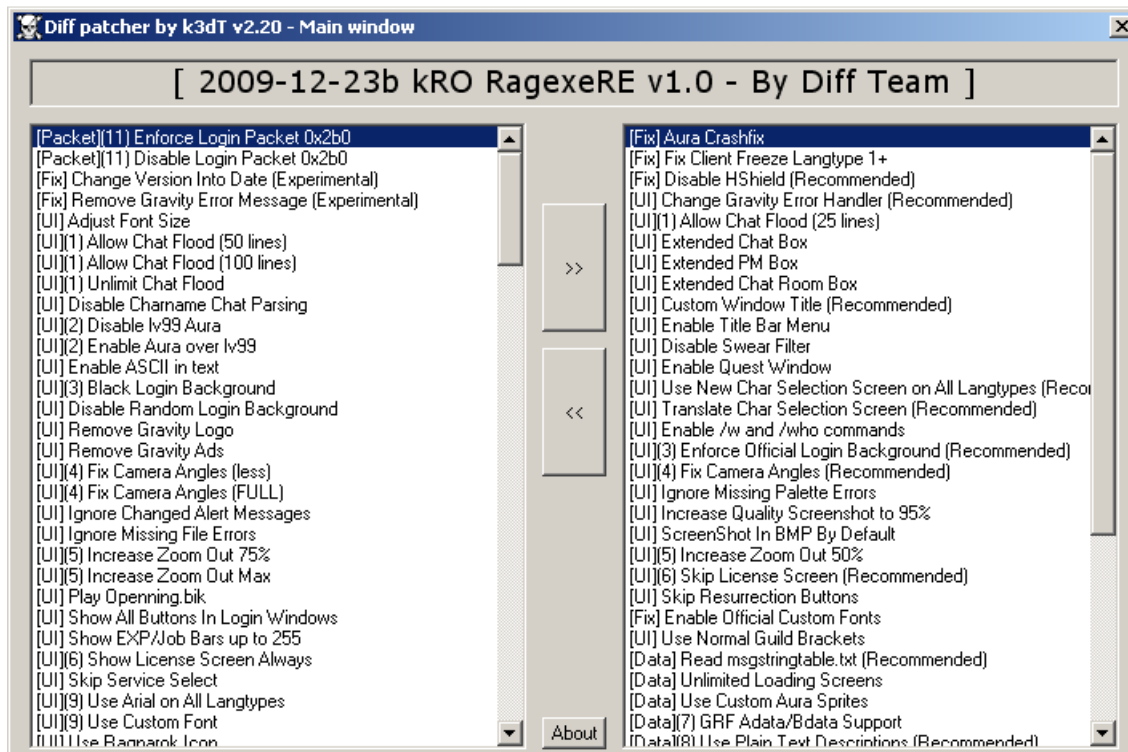
En realidad se trata de dos clientes con sus correspondientes instaladores, **Ragnarok** y **Renewal**<sup>[95]</sup>, pero se engloba en uno porque ambos son indispensables para jugar. Son oficiales y están disponibles para descargarse de forma gratuita, ya que lo que se compra a Gravity es la suscripción mensual al juego.

El Ragnarok es la base del juego y contiene todo lo necesario para hacerlo funcionar, mientras que el Renewal es una ampliación necesaria para jugar, ya que el servidor está diseñado específicamente para ella. Antiguamente la ampliación utilizada se llamaba **Sakray**, y todavía algunos servidores desactualizados la utilizan (no existe ningún servidor que permita jugar sin una de esas dos ampliaciones). Desde 2008, en los servidores coreanos se cambió radicalmente el sistema de juego y todo lo que decidieron mantener del Sakray pasó al cliente Ragnarok, creando el cliente Renewal para implementar en él todas las novedades posteriores. Nótese que se ha utilizado el término *ampliación* para definir de forma más clara al Renewal y al Sakray, pero en todas partes se los conoce como *clientes*, y no existen más que esos dos.

### 7.3.2 **Cliente BidaiRO**

Todo servidor de Ragnarok tiene un cliente personalizado que permite a sus usuarios acceder al juego. Del cliente oficial de Gravity sólo se utilizan los datos (imágenes, sonidos, algunas configuraciones...), no los ejecutables, ya que estos intentarían conectarse al servidor oficial. En su lugar se utiliza un ejecutable personalizado que se crea aplicando un parche al ejecutable del Renewal para

modificarlo con las características que se quieran obtener. Existen programas para aplicar los parches de forma automática, pero el más conocido y utilizado es el **Diffpatcher**<sup>[96]</sup> de **k3dT**. Los parches los mantiene el **Diffteam**<sup>[168]</sup>, un grupo de desarrolladores que se ocupan de mantenerlos al día debido a las numerosas actualizaciones que sufre el cliente Renewal. La interfaz del Diffpatcher es muy sencilla e intuitiva, como puede verse en la *Imagen 24: Ventana principal de Diffpatcher*.



**Imagen 24: Ventana principal de Diffpatcher**

La información de los parches se obtienen mediante ficheros donde se indica el offset<sup>[97]</sup> a modificar con sus bytes correspondientes, y el tipo de modificación que será para que la muestre el Diffpatcher, es decir, se altera directamente el binario ya que no se dispone de su código fuente. Aparte de estas modificaciones automáticas, es necesario realizar otra manualmente para inyectarle al cliente una DLL donde se encuentran ciertas funciones necesarias para el anticheat. Dado que la creación de clientes es un proceso que se lleva a cabo muy pocas veces, este proceso no está automatizado por el momento. Los pasos se pueden ver gráficamente en el *Anexo F: Pasos para la creación de un cliente*.

Para el despliegue del cliente se ha creado un instalador con su correspondiente desinstalador mediante el software **NSIS**<sup>[98][169]</sup>, donde se incluye, aparte del cliente, las DLL necesarias para la seguridad del servidor (anticheat y encriptación<sup>[99]</sup> de paquetes<sup>[100]</sup>), un parcheador<sup>[101]</sup> para mantener el cliente siempre actualizado, ficheros de datos con información de conexión y varias personalizaciones para el juego de tipo gráfico, de audio, traducciones y similares y un mp3 con una canción con licencia Creative Commons<sup>[170]</sup> que se escucha al empezar a jugar.

### 7.3.3 GRFs

El **GRF**<sup>[102]</sup> es un fichero comprimido utilizado por Ragnarok Online. Dentro podría guardarse cualquier tipo de información, pero se utiliza para almacenar sonidos, música de corta duración, gráficos, textos y algún fichero de configuración (por ejemplo para indicar a qué servidor debe conectar el cliente). Todo lo que se almacena en estos ficheros es para uso exclusivo del cliente, de forma que cuando el jugador utiliza una habilidad los efectos son obtenidos del GRF y mostrados en pantalla. Igualmente, cuando el servidor se comunica con el cliente para informarle de lo que acontece, este escoge los datos necesarios del GRF para mostrárselo al jugador.

Existen dos GRF principales que todo servidor de Ragnarok tiene, tanto si es oficial como si es privado: **data.grf** y **rdata.grf**. El primero viene con la instalación del Ragnarok, ocupa más de **1.3GB**<sup>[103]</sup> (con cada actualización aumenta de tamaño) y en él se encuentran todas y cada una de las cosas que se pueden ver y oír a través del cliente: los mapas, las ciudades, los personajes, los efectos de sonido, etc. El segundo se instala con el Renewal, y su contenido es similar al anterior pero con información exclusiva de esa expansión. Ocupa más de **170MB**<sup>[104]</sup>, muy poco comparado con el data.grf, pero crece a un ritmo mayor ya que las actualizaciones tienen más contenido Renewal.

Además de estos GRF, cada servidor privado puede tener uno o más GRFs particulares, donde como mínimo se almacenará la información de conexión y algunas traducciones, ya que el juego original está en coreano. También es posible tener todos estos datos en una carpeta que leerá el cliente al iniciarse (puede definirse este comportamiento al crear el cliente con el Diffpatcher), pero es más sencillo actualizar



un GRF y también ofrece mayor seguridad, ya que es posible controlar que no se ha modificado con mayor facilidad.

En BidaiRO existen dos GRFs de uso público y otro de uso privado para los administradores. El primero, **bidai.grf**, de unos 21kB, contiene la información que necesita el cliente para conectar al servidor, traducciones de todos los nombres y mensajes que se pueden mostrar en el juego (la mayoría en inglés, ya que existen packs<sup>[171]</sup> predefinidos, pero algunos han sido traducidos al español), varios gráficos para mostrar distintos tipos de ropa, peinados y color de pelo en los personajes a elegir por el jugador, algunos sonidos e información sobre ítems decorativos personalizados que no están disponibles en el servidor oficial.

El segundo, **bidai2.grf**, de unos 4kB<sup>[105]</sup>, es un concepto exclusivo del servidor y permite personalizar el aspecto gráfico del juego de forma individual para jugador, a su elección. Contiene imágenes y algunos textos que se descargan desde la web según se quieran utilizar, de forma que si un día se quieren visualizar los nombres de las habilidades en español, se reemplaza la información del GRF con la correspondiente de la web, y si otro día quieren verse en inglés, se realiza el mismo proceso a la inversa.

El GRF de uso privado, **bidgm.grf**, simplemente contiene la dirección del servidor de pruebas para poder conectarse a él además de al público. Es necesario un cliente especial para administradores, ya que los GRF que se van a leer se especifican al crear cada cliente.

Los GRF se procesan en orden y esto es importante, ya que la información se sobrescribe y es lo que permite que puedan aplicarse tantas personalizaciones. El orden se muestra en la *Tabla 4: Orden de carga de los GRF*.

Orden	Fichero
1º	data.grf
2º	rdata.grf
3º	bidai.grf
4º	bidai2.grf

Tabla 4: Orden de carga de los GRF

Esto quiere decir que para traducir algo basta con añadir un fichero a bidai.grf con los textos traducidos en la misma ruta y con el mismo nombre que el que haya en data.grf (o rdata.grf, pero es más común que esté en el primero). Como se leerá después, encontrará el fichero y lo reemplazará por el contenido personalizado. En esto se basa el funcionamiento de bidai2.grf, el usuario elige qué fichero va a escribir en él y este sustituye al que se encuentre en data.grf, pudiendo personalizar el aspecto.

Además de crear nuevos GRF donde añadir cualquier personalización, es posible modificar uno ya existente como se hace con el bidai2.grf. Esto puede ser utilizado por los jugadores para hacer trampas, por ejemplo modificando en el data.grf el color que se utiliza para mostrar que un personaje ha sido cegado (hace que casi toda la pantalla se vea negra). Cambiando este negro por un color transparente es posible anular su efecto. Para evitar esto, se han duplicado en el bidai.grf los ficheros susceptibles de ser modificados para estos fines, y se hace un chequeo del CRC mediante el anticheat para comprobar que no se ha cambiado. Así aunque un usuario modifique el original en data.grf, en realidad estará usando el de bidai.grf, y si modifica este último el CRC será incorrecto y no podrá entrar al juego. No es viable comprobar el CRC de data.grf ni de rdata.grf, primero por el tamaño de los ficheros, y segundo porque cada usuario puede actualizarlo a placer y sería imposible tener una lista de tantos CRC (se actualizan diariamente). En cambio el bidai.grf tiene un tamaño mucho más asequible y las actualizaciones están controladas. En el caso del bidai2.grf, cuyo contenido puede variar a discreción del usuario, se calcula el CRC de las rutas de los ficheros, que son siempre las mismas. De esta forma si se añade alguna cosa el CRC será incorrecto, y si algún usuario lo modifica con sus propios ficheros no pasará nada porque no supone ningún riesgo.

Para trabajar con los GRF existen varias aplicaciones, pero en BidaiRO se utilizan principalmente dos: **GRF Tool**<sup>[172]</sup> y **Gryff**<sup>[173]</sup>, ambas gratuitas. La primera es un editor de GRFs que permite visualizar su contenido y tiene opciones de previsualización de imágenes, textos y reproducción de sonidos, además de ofrecer la opción de extraer todo o parte del contenido del fichero y de filtrarlo para ver sólo una parte del mismo. Se muestra su interfaz en la *Imagen 25: Bidai.grf visto desde GRF Tool*.

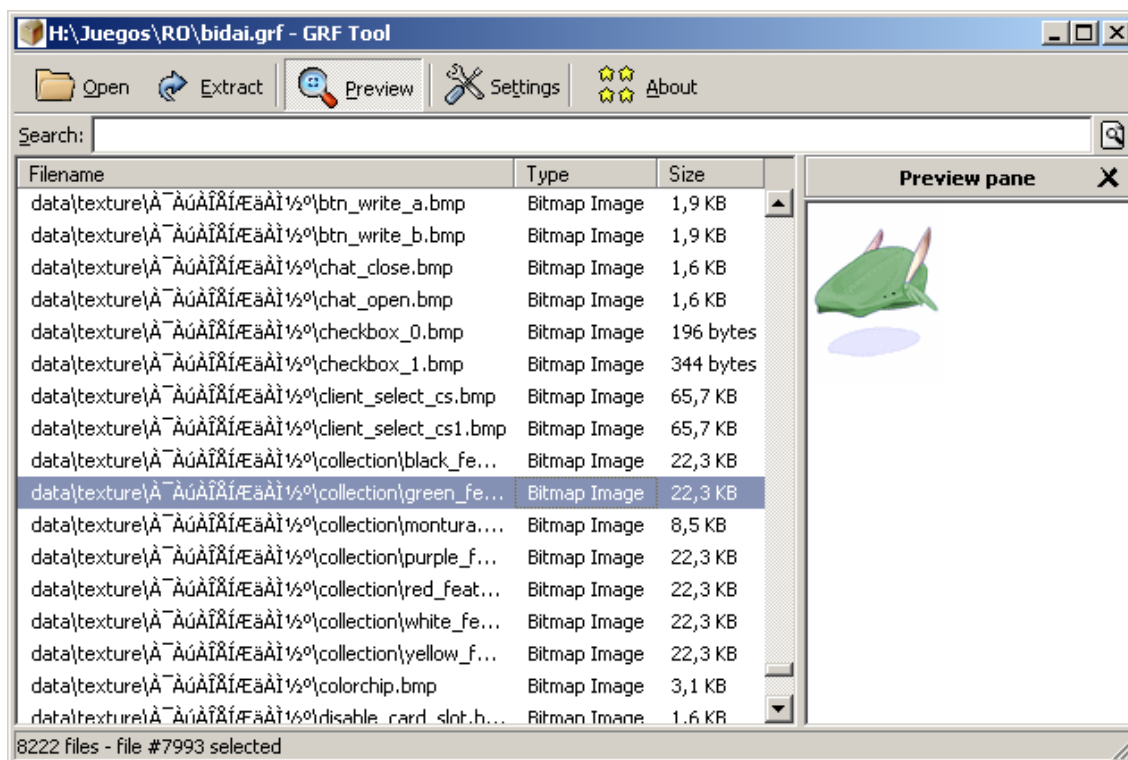


Imagen 25: Bidai.grf visto desde GRF Tool

La segunda aplicación, **Gryff**, cuya interfaz se puede ver en la *Imagen 26*: *Parche editado con Gryff*, es un creador de GRFs que permite crearlos desde cero, modificar el contenido de uno existente o unir dos en uno solo. La primera opción se utiliza para crear parches con los que se podrán actualizar otros GRF, y el resto es útil para probar los parches antes de hacerlos públicos o para probar cualquier otra cosa que requiera modificar un GRF.

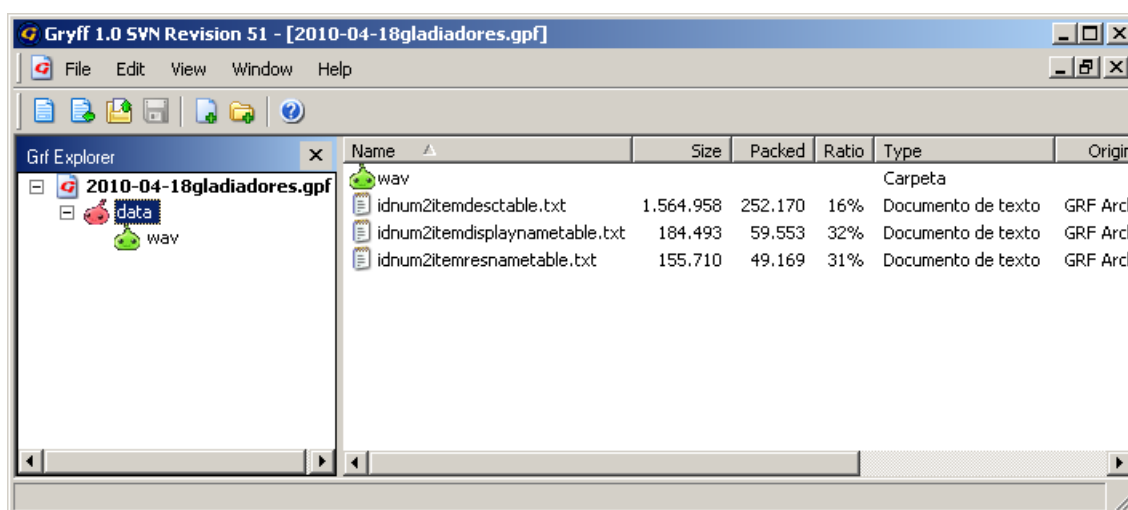


Imagen 26: Parche editado con Gryff

Para diferenciar un parche del GRF principal, por convención se cambia la extensión de estos ficheros a **GPF**, aunque la estructura sea la misma.

Además de esas aplicaciones se cuenta con otras dos realizadas en **C#**<sup>[108]</sup> con librerías **.NET**<sup>[109]</sup> para aplicar los parches y calcular los CRC, implementadas y utilizadas por la administración del servidor. En la *Imagen 27: Aplicación privada para parchear* y la *Imagen 28: Aplicación privada para calcular CRCs* se puede ver sus interfaces, muy básicas ya que son de uso interno e implementan lo mínimo indispensable.

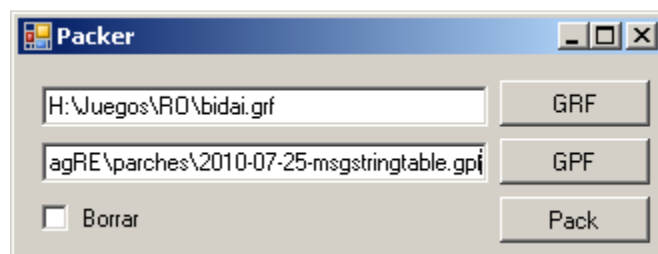


Imagen 27: Aplicación privada para parchear

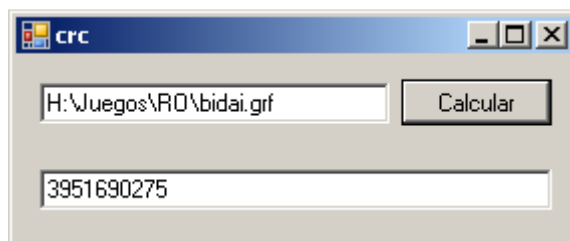


Imagen 28: Aplicación privada para calcular CRCs

#### 7.3.4 Parcheador

El parcheador se utiliza para mantener al día los componentes más importantes del cliente de BidaiRO (no se aplica para el Ragnarok y el Renewal, que tienen sus propios parcheadores de Gravity). Los ficheros que puede actualizar se muestran en la *Tabla 5: Ficheros actualizables a través del parcheador*.

Nombre	Descripción
<b>BidaiPatch.exe</b>	El propio parcheador.
<b>bgl.dll</b> ( <b>BidaiRO GRF Library</b> )	DLL utilizada por el parcheador para realizar el tratamiento de los GRFs.
<b>BidaiRO.dll</b>	DLL utilizada por el cliente para llevar a cabo la encriptación de los paquetes.
<b>anticheat.dll</b>	DLL utilizada por el cliente para calcular los CRCs y comunicarse con el servidor anticheat.
<b>BidaiRO.exe</b>	Ejecutable del cliente.
<b>bidai2.grf</b>	GRF de personalizaciones del usuario.
<b>bidai.grf</b>	GRF principal de BidaiRO.

**Tabla 5: Ficheros actualizables a través del parcheador**

El parcheador está desarrollado en **C#** con librerías **.NET**, y utiliza la librería **bgl.dll** que contiene funcionalidades para agregar y borrar ficheros de forma individual de un GRF. Esta librería está programada en **C** y para su desarrollo fue necesario un importante estudio del sistema de compresión GRF. En el *Anexo G: Implementación del parcheador*, se hace un análisis del código.

Hoy en día el parcheador se utiliza sólo para comprobar si hay actualizaciones y para dar la opción de personalizar ciertos aspectos visuales del juego. Al iniciarse, se conecta con la web y obtiene un fichero con las versiones de cada componente, que se compara con una copia local donde está guardada la última versión descargada. El parcheador se descargará aquellos componentes cuyas versiones en la máquina local sean inferiores a las del servidor, y dependiendo de qué tipo de fichero sea realizará unas acciones u otras, como puede observarse en la *Tabla 6: Acciones realizadas por el parcheador dependiendo del fichero*.

Acción	Ficheros
<b>Sustitución y reinicio</b>	BidaiPatch.exe bgl.dll
<b>Sustitución</b>	BidaiRO.dll anticheat.dll BidaiRO.exe
<b>Parcheo (añadir/borrar/modificar)</b>	bidai.grf bidai2.grf

Tabla 6: Acciones realizadas por el parcheador dependiendo del fichero

Antiguamente, cuando el anticheat todavía no estaba implementado en el servidor, se utilizaba el parcheador para detectar qué jugadores podían estar utilizando algún fichero modificado y así observarlos para tratar de detectar si hacían algo ilegal. Para ello se enviaba al servidor el CRC de algunos ficheros y se almacenaban en la base de datos junto con alguna información extra, para después generar un informe mediante un proceso de minería de datos que se enviaba diariamente por correo electrónico a los administradores del servidor, todo de forma automática. Esta comunicación se realizaba mediante el servidor **Apache** utilizando **Python** con el módulo **Mod\_python**<sup>[174]</sup>, necesario para poder utilizar el lenguaje dentro del servidor web. Tras la implementación del servidor anticheat, estas comprobaciones quedaron obsoletas y se deshabilitaron.

Es obligatorio utilizar el parcheador para entrar al servidor. El ejecutable del cliente no se puede abrir haciendo doble clic sobre él (aunque es muy sencillo saltarse esto y también estaba contemplado en el antiguo anticheat, pero no representa ningún problema en cuanto a seguridad), de forma que todos los usuarios se aseguran de tener siempre la última versión del cliente, o de lo contrario no podrían entrar al juego. Por eso es importante que al subir cualquier parche se actualicen en el anticheat los CRCs que hayan cambiado, o de lo contrario nadie podría acceder al servidor. Los CRCs viejos normalmente se dejan unos días antes de borrarlos, por si algún usuario tuviera cualquier problema con la actualización o se hubiera realizado de forma incorrecta.

A continuación se muestra la *Imagen 29: Parcheador con la ventana de personalización*, donde se observan las opciones para jugar, acceder a la web principal, al foro y a la página de registro, además de la opción de personalizar los aspectos visuales. Al iniciar realiza una conexión a la web del servidor que además de obtener las versiones de los parches, muestra una página con enlaces a **Google AdSense**<sup>[110][175]</sup> y a la página de votaciones en los tops del panel de control, así como las secciones de **anuncios**, **noticias** y **actualizaciones** que también pueden verse desde la web y el foro.



Imagen 29: Parcheador con la ventana de personalización

En el *Anexo H: Ficheros utilizados por el parcheador* puede verse un ejemplo del fichero que descarga el parcheador con las versiones de los distintos componentes y del fichero local con el que comprueba si es necesario actualizar.

## 7.4 Servidor MySQL

Dada la cantidad de datos que se manejan en el servidor y la interacción que hay entre sus componentes, es crucial el uso de un buen gestor de bases de datos correctamente configurado. Para ello se utiliza **MySQL 5**<sup>[176]</sup> en su versión gratuita, que proporciona todas las funcionalidades necesarias. También se pensó en utilizar **PostgreSQL**<sup>[177]</sup>, ya que al menos en el momento del análisis ofrecía triggers más potentes, pero se escogió MySQL porque la mayoría de los componentes que forman el servidor están pensados para utilizar ese gestor.

La configuración básica del servidor MySQL ha sido modificada para adaptarla a las necesidades de BidaiRO y optimizar el rendimiento en base a varias estimaciones de uso<sup>[178]</sup>. En el *Anexo I: Configuración del servidor MySQL*, puede verse el fichero de configuración del servidor.

Además de eso, periódicamente se ejecutan dos scripts para, si es necesario, ajustar la configuración en base a los resultados obtenidos o realizar tareas de mantenimiento. Los scripts utilizados son **tuning-primer.sh**<sup>[179]</sup> de Matthew Montgomery y **mysqltuner.pl**<sup>[180]</sup> de Major Hayden. En el *Anexo J: Análisis de rendimiento de MySQL*, pueden ver unos ejemplos de su utilización.

### 7.4.1 Bases de datos y tablas

Cada componente del servidor, a excepción de Apache, utiliza bases de datos para funcionar. Es por ello que MySQL se ocupa de gestionar varias bases de datos diferentes con sus correspondientes tablas. A continuación se muestran las principales en la *Tabla 7: Listado de bases de datos utilizadas*.

Base de datos	Descripción
<b>ragnarok</b>	Base de datos principal del emulador <b>eAthena</b> , se almacena todo lo relativo al juego (cuentas, personajes, clanes, monstruos, objetos, etc.)



<b>ragnarok_log</b>	Base de datos de logs <sup>[111]</sup> del emulador <b>eAthena</b> , se almacenan logs de distintos tipos (conexiones realizadas, intercambios de objetos, comandos utilizados, etc.)
<b>ragnarok_foro</b>	Base de datos del foro <b>phpBB</b> , contiene todo lo necesario para su funcionamiento (usuarios, grupos, foros, publicaciones, configuraciones, etc.)
<b>ragnarok_cp</b>	Base de datos del panel de control <b>Ceres</b> , almacena autenticaciones con clave incorrecta y un log de las consultas realizadas a la base de datos.
<b>bidai_mobs</b>	Base de datos propia de <b>BidaiRO</b> , almacena sets de monstruos para poder invocarlos mediante comandos o scripts y realizar eventos cómodamente.
<b>log_history</b>	Base de datos propia de <b>BidaiRO</b> , replica la información de <i>ragnarok_log</i> mediante un trigger cada vez que se insertan datos, ya que estos se borran periódicamente para aligerar las copias de seguridad. Teniéndolos en una base de datos aparte pueden consultarse cómodamente y sin provocar sobrecarga en el juego al estar en una base de datos distinta.
<b>ts</b>	Base de datos de <b>TeamSpeak</b> , almacena toda la información necesaria para su funcionamiento (cuentas, canales, permisos, etc.). Hay que destacar que TeamSpeak por defecto utiliza <i>SQLite</i> <sup>[112][181]</sup> y hubo que realizar una conversión y varios ajustes para poder utilizarlo con <i>MySQL</i> , algo que se consideró positivo teniendo en cuenta que el resto de componentes también lo utilizaban.
<b>dev_ragnarok</b>	Réplica de la base de datos <i>ragnarok</i> para el entorno de <b>desarrollo</b> .
<b>dev_ragnarok_log</b>	Réplica de la base de datos <i>ragnarok_log</i> para el entorno de <b>desarrollo</b> .

Tabla 7: Listado de bases de datos utilizadas

La mayoría de las bases de datos son las necesarias para que funcione su componente correspondiente, por lo que se listarán sólo las tablas que han sido añadidas para implementar otras funcionalidades en la *Tabla 8: Tablas de BidaiRO para la base de datos RAGNAROK*, *Tabla 9: Tablas de BidaiRO para la base de datos RAGNAROK\_LOG*, *Tabla 10: Tablas de BidaiRO para la base de datos RAGNAROK\_FORO* y *Tabla 11: Tablas de BidaiRO para la base de datos TS*.

RAGNAROK	
Tabla	Descripción
<b>anticheat</b>	Almacena los CRCs de los distintos componentes que se analizan con el anticheat. Cuando un usuario quiere entrar al juego se comparan sus CRCs con los presentes en esta tabla, que puede actualizarse de forma dinámica para añadirlos o quitarlos en caliente.
<b>apuestas_cuentas</b>	Esta tabla aún no se utiliza, fue creada para implementar un proyecto de casa de apuestas dentro del juego, de forma que cada jugador pueda apostar una cantidad de dinero a algún tema propuesto y los ganadores se llevan el bote. En esta tabla irá la información relativa a la apuesta realizada por cada cuenta de usuario.
<b>apuestas_temas</b>	Al igual que la tabla anterior, todavía no se está utilizando. Aquí se almacenarán los temas disponibles sobre los que apostar y toda la información necesaria, como el bote acumulado, la fecha de finalización, el precio de cada apuesta, etc.
<b>check_dupes</b>	Para controlar el duplicado ilegal de ítems por parte de los jugadores se ha creado un sistema (todavía por depurar) donde se almacena el total de cada ítem existente en el servidor y se comprueba periódicamente con el total que tienen los jugadores. Aquí se almacenan los totales existentes. Cada vez que un jugador obtiene o pierde un ítem se guarda la información en los logs de eAthena, y mediante un trigger es posible añadir y restar los totales en esta tabla.

<b>conquistas</b>	Las <i>conquistas</i> son un evento diario de BidaiRO donde varios clanes pueden luchar entre sí para hacerse con el control de una ciudad de una forma única y novedosa. En esta tabla se guarda información relativa a esas ciudades, como el clan que la controla, el dinero almacenado en sus arcas, la fortaleza de sus defensas, etc.
<b>conquistas_guardianes</b>	Es posible contratar guardianes para las ciudades conquistadas. En esta tabla se almacena toda la información relativa a estos guardianes, como el tipo de guardián, la ciudad a la que pertenece, si patrulla la ciudad o guarda una zona, etc.
<b>encuestas</b>	BidaiRO dispone de un sistema de encuestas dentro del propio juego, donde puede exponerse un tema durante una fecha determinada con distintas opciones para responder. En esta tabla se almacena esa información y las respuestas obtenidas.
<b>max_usuarios</b>	Tabla para almacenar el máximo de usuarios conectados cada día, así como la media relativa a los días anteriores.
<b>multiplay_whitelist</b>	En BidaiRO no se permite más de una conexión desde la misma IP para evitar que un jugador juegue al mismo tiempo con más de un personaje. En esta tabla se almacenan las cuentas que pueden saltarse esta restricción para permitir casos de hermanos que jueguen desde la misma casa o similares.
<b>multiplay_whitelist2</b>	Similar a la anterior, pero se crean grupos de cuentas que puedan jugar al mismo tiempo. Útil cuando hay que habilitar varias cuentas asociadas entre ellas para no tener que hacer el producto cartesiano <sup>[113]</sup> de todas ellas.
<b>whitelist2_descriptions</b>	Descripciones de cada grupo creado para la tabla <i>multiplay_whitelist2</i> , de forma que sea sencillo gestionarlos.
<b>ranking_break</b>	Tabla para almacenar las veces que cada clan ha conseguido un castillo (para lo cual hay que romper el Emperium, de ahí el nombre de la tabla) y así poder mostrarlo en un ranking.

<b>ranking_castle</b>	Tabla para almacenar las veces que cada clan ha finalizado la WoE con un castillo. No es lo mismo que la anterior, ya que un clan puede conseguir un castillo pero perderlo y finalizar la WoE sin él (conservar el castillo es lo que da beneficios).
<b>ranking_pvp99</b>	BidaiRO tiene un sistema de combate PvP en el que cada jugador puede crearse un personaje de nivel máximo que se traslada a una sala de la que no puede volver a salir (para no interferir en el juego normal), donde puede elegir el job que quiera las veces que quiera y equiparse una amplia variedad de equipo. Tres veces a la semana se organizan de forma automática combates entre estos personajes, y en esta tabla se almacenan las estadísticas de los combates.
<b>sospechosos</b>	En esta tabla se almacenan las cuentas de jugadores que podrían estar haciendo algo ilegal, junto con el motivo. Cada vez que algún personaje de esas cuentas entra al juego, se muestra un mensaje visible sólo para los administradores informándolo (tanto en las cuentas de administrador como las de jugador normal asociadas), de forma que se puedan tomar las medidas necesarias en el menor tiempo posible.
<b>trivial</b>	BidaiRO cuenta con un trivial dentro del juego que se organiza una vez a la semana de forma automática (aunque es posible arrancarlo manualmente). En esta tabla se almacenan todas las preguntas del trivial junto con sus respuestas, el tema al que pertenecen y su autor. Actualmente hay disponibles 135.858 preguntas.
<b>trivial_puntuaciones</b>	Aquí se guarda la puntuación obtenida por cada jugador en el último trivial realizado. Es necesario guardarla en una tabla para que en caso de que el jugador reconecte durante la partida no se pierda.
<b>viciometro</b>	En esta tabla se almacena el tiempo de juego de cada personaje por día jugado, de forma que se pueden realizar estadísticas diarias o totales.

<b>votepoints</b>	Guarda la última vez que ha votado cada jugador en los cuatro tops, de forma que pueda comprobarse si ha pasado el tiempo mínimo establecido antes de asignar los votepoints.
-------------------	---

Tabla 8: Tablas de BidaiRO para la base de datos RAGNAROK

RAGNAROK_LOG	
Tabla	Descripción
<b>anticheatlog</b>	Cada vez que un jugador intenta acceder al juego, se almacena en esta tabla toda la información sobre su cuenta, fecha, hora de acceso, CRCs validados, resultados de las validaciones y rutas de los componentes en su equipo.
<b>contestador</b>	Se ha diseñado un sistema de contestador automático para que los jugadores tengan otra forma de comunicarse con los administradores desde el juego. Basta con hablarles por privado y si están desconectados, el mensaje se almacena en esta tabla con varios datos del remitente, para después enviarse por correo al administrador correspondiente todos los días por la mañana.
<b>multiplay</b>	En esta tabla se almacenan los accesos simultáneos de cuentas desde la misma IP, de forma que con un poco de minería de datos se puede obtener una lista de jugadores con altas probabilidades de estar utilizando estos permisos de manera ilegal.
<b>trivial_log</b>	Información general de cada trivial que se almacena una vez ha finalizado, como la fecha y hora de inicio y de fin, el número de preguntas que tenía, los ganadores y los premios repartidos, etc.
<b>trivial_ranking</b>	Tabla para almacenar el total de puntos, respuestas acertadas y premios que ha obtenido cada jugador en un trivial. Todo esto podría obtenerse de <i>trivial_log</i> , pero sería más lento y no conviene teniendo en cuenta que esta consulta se realiza a menudo desde la web.

<b>trivial_respuestas</b>	Aquí se guarda el tiempo que tarda cada jugador en contestar a una pregunta. De esta forma es posible obtener una lista de gente que puede servirse de Google o un buscador similar para obtener las respuestas.
<b>votepoints_log</b>	Log donde se almacena el número de votos que recibe cada top en un día.

Tabla 9: Tablas de BidaiRO para la base de datos RAGNAROK\_LOG

RAGNAROK_FORO	
Tabla	Descripción
<b>foro_ro</b>	Tabla para guardar la correspondencia entre la cuenta del foro y la cuenta del juego de los jugadores.

Tabla 10: Tablas de BidaiRO para la base de datos RAGNAROK\_FORO

TS	
Tabla	Descripción
<b>ro_ts</b>	Tabla para guardar la correspondencia entre la cuenta del juego y la cuenta del TeamSpeak de los jugadores.

Tabla 11: Tablas de BidaiRO para la base de datos TS

Está prevista una reestructuración de algunas tablas, como por ejemplo las de log de la base de datos *ragnarok*, que estarían mejor en *ragnarok\_log*. También, aparte de las tablas aquí mostradas, se han añadido en algunos casos campos a otras tablas originales de eAthena; la tabla **login** por ejemplo, donde se almacenan las cuentas de los usuarios, contiene algún campo relativo al anticheat, votepoints o similar. La tabla **char**, donde se almacenan los personajes de los jugadores, contiene varios campos relativos generalmente a rankings. En estos casos habría sido mejor crear otras tablas con el nuevo contenido, asociadas a las originales mediante la clave primaria<sup>[114]</sup>, de forma que la implementación de eAthena quede lo más separada posible de la de BidaiRO. Este es un punto que también está previsto cambiar.

### 7.4.2 Usuarios y permisos

Para utilizar las distintas bases de datos se han creado una serie de usuarios con los permisos mínimos necesarios para realizar las funciones que requiera su rol. De esta forma se gana en seguridad, ya que si alguien se hace con la contraseña de uno de esos usuarios o descubre alguna vulnerabilidad el acceso que tendrá a la base de datos será limitado.

También es más seguro para la gestión interna del servidor. Por ejemplo, si se quiere restaurar una copia de seguridad en el servidor de desarrollo, se cargará con el usuario de la base de datos de desarrollo, que no tiene permisos en el entorno de producción. Gracias a esto si por un descuido se intenta cargar la copia en la base de datos de producción no ocurrirá nada porque el usuario no tendrá permisos.

Otra de las ventajas de tener varios usuarios con permisos bien definidos es que es mucho más sencillo mantener todo el sistema. Por ejemplo, consultando los permisos del usuario principal de cualquier base de datos se puede ver fácilmente qué tipo de interacción existe con las demás

Los usuarios y sus roles son los siguientes:

- **administracion:** Posiblemente el usuario más utilizado debido a que tiene sólo permisos de lectura en todas las bases de datos, salvo casos concretos en los que puede llegar a realizar alguna modificación o ejecutar algún procedimiento almacenado<sup>[115]</sup> o función<sup>[116]</sup>, todos ellos para tareas administrativas. Gracias a esto es posible realizar cualquier consulta sin temor a modificar nada.
- **ragnarok:** Usuario principal de las bases de datos *ragnarok* y *ragnarok\_log*. Cuenta con prácticamente todos los permisos para esas bases de datos, aunque en la práctica el usuario *dev\_ragnarok* para el entorno de desarrollo, descrito más abajo, tiene alguno más para su base de datos. También tiene permisos de lectura en *bidai\_mobs* y de inserción en *log\_history*, para poder actualizarla mediante los triggers, así como permisos limitados de lectura e inserción en *ragnarok\_foro*,

necesarios para poder obtener los anuncios que se mostrarán en el juego y para poder escribir reportes de errores en el bugtracker, que se almacena como una publicación más dentro de la base de datos del foro, con alguna diferencia.

- **foro:** Usuario principal de la base de datos *ragnarok\_foro*, para la que tiene permisos de lectura, inserción, actualización y borrado. También necesita permisos de selección e inserción en algunas tablas de las bases de datos *ragnarok* y *ts*, para poder crear las cuentas de los usuarios desde la web (aunque no se haga desde el propio foro se utiliza el mismo usuario).
- **cp:** Usuario principal de la base de datos *ragnarok\_cp*, tiene permisos de lectura, inserción y actualización en esta base de datos. Como desde el panel de control se pueden realizar algunas modificaciones en la propia cuenta de usuario y se gestionan los votepoints, que están en la base de datos de logs, tiene un mínimo de permisos en las bases de datos *ragnarok* y *ragnarok\_log* que le permiten acceder y modificar ciertas tablas.
- **ts:** Usuario principal de la base de datos *ts*, cuenta con todos los permisos necesarios para gestionarla.
- **vote:** Usuario encargado de gestionar los votepoints desde el foro. La información de los votos se almacena en las cuentas de cada usuario, por lo que tiene permisos restringidos de selección, inserción y actualización en la base de datos *ragnarok*.
- **backup:** Este usuario se ejecuta de forma automática desde varios scripts de Linux y tiene permisos de lectura y bloqueo de tablas en varias bases de datos, para realizar la copia de seguridad de las mismas. Dado que los logs antiguos se eliminan progresivamente para no retardar este proceso, también necesita permisos de borrado y alteración de tablas en la base de datos *ragnarok\_log*.
- **dev\_ragnarok:** Usuario principal de las bases de datos *dev\_ragnarok* y *dev\_ragnarok\_log*, para las que tiene todos los permisos. Es el usuario que se utiliza en el servidor de desarrollo, y también tiene permisos de



lectura en *bidai\_mobs* y *ragnarok\_foro* (necesarios para mostrar los anuncios del foro dentro del juego).

En la tabla *Tabla 12: Usuarios y permisos de las bases de datos*, puede verse un resumen de los permisos de cada usuario para todas las bases de datos del servidor. Las celdas coloreadas en azul indican un acceso total a la base de datos, mientras que las naranjas indican un acceso parcial, es decir, sólo se tiene el permiso correspondiente en una o más tablas de esa base de datos, pero nunca en todas. Las celdas en blanco indican que el usuario no tiene ese permiso en la base de datos.

		dev_ragnarok_log	dev_ragnarok	ts	log_history	bidai_mobs	ragnarok_cp	ragnarok_foro	ragnarok_log	ragnarok
administracion	SELECT									
	INSERT									
	UPDATE									
	DELETE									
	EXECUTE ON PROCEDURE									
	EXECUTE ON FUNCTION									
ragnarok	SELECT									
	INSERT									
	UPDATE									
	DELETE									
	CREATE									
	DROP									
	INDEX									
	ALTER									
	CREATE TEMPORARY TABLES									
	LOCK TABLES									
	EXECUTE ON FUNCTION									
foro	SELECT									
	INSERT									
	UPDATE									
	DELETE									
cp	SELECT									
	INSERT									
	UPDATE									
	DELETE									
	EXECUTE ON FUNCTION									

	ts	vote	backup	dev_ragnarok
	SELECT			
	INSERT			
	UPDATE			
	DELETE			
	CREATE			
	DROP			
	EXECUTE			
	SELECT			
	UPDATE			
	INSERT			
	SELECT			
	DELETE			
	ALTER			
	LOCK TABLES			
	ALL			
	SELECT			

**Tabla 12: Usuarios y permisos de las bases de datos**

## **8 Pautas para la gestión**

## 8.1 Pautas para la gestión

En todo proyecto es importante definir ciertas pautas para gestionarlo, ya sea a nivel interno como a nivel atención al usuario. Es necesario cuantificar todo lo posible para tener claro qué servicios van a poder ofrecerse y con qué limitaciones, definir políticas para gestionar el día a día y preparar un plan de acción ante eventuales incidencias. A continuación veremos algunas de las pautas definidas para el servidor BidaiRO.

### 8.1.1 Gestión de capacidad

Es necesario saber cuántos recursos consume el servidor con un cierto número de usuarios para poder estimar el máximo que puede admitir. Para ello, en la *Tabla 13: Consumo de recursos del servidor* se analizará el porcentaje de memoria RAM<sup>[117]</sup>, CPU<sup>[118]</sup>, disco duro y ancho de banda consumido por los servidores más importantes, con 50 usuarios y con 100.

Usuarios	Recurso	Map server	MySQL	Apache	Total
50	RAM	13%	10%	3.2%	26.2%
	CPU	5%	2%	1%	8%
	Disco duro	-	3.4%	-	3.4%
	Ancho de banda	-	-	-	2%
100	RAM	18%	13%	3.9%	34.9%
	CPU	7%	3%	1%	11%
	Disco duro	-	3.6%	-	3.6%
	Ancho de banda	-	-	-	4%

Tabla 13: Consumo de recursos del servidor

Como se puede observar, el servidor tiene capacidad más que de sobra para 100 usuarios. El recurso más consumido es la memoria, por lo que se realizará una estimación del máximo de usuarios en base al aumento reflejado en la tabla. De 50 a 100 usuarios se aprecia una diferencia de 8.7 puntos, por lo que es previsible que el

servidor alcance su capacidad máxima con **474** usuarios (calculando un 8.7% de aumento por cada 50 usuarios más).

Los recursos totales del servidor se podrán consultar más adelante en la *Tabla 15: Características del host (máquina y red)*.

### **8.1.2 Escalabilidad**

Tras el análisis de capacidad realizado anteriormente, se puede establecer un umbral de **400 usuarios** hasta que se considere necesario plantearse aumentar las características de la máquina que aloja al servidor. Una vez alcanzados los 400 usuarios, deberá procederse a revisar el consumo de recursos y compararlo con el consumo estimado. Si se acerca a la estimación, se considerará seguro aumentar el umbral a 450.

Es posible que la estimación no sea correcta, de ahí que se deba fijar el umbral a un valor más bajo del estimado. El consumo de CPU por parte del *Map server* es lo que más se puede disparar, ya que a mayor cantidad de usuarios, mayor interacción habrá entre ellos y mayores cálculos tendrá que hacer el servidor, por lo que probablemente aumente de forma exponencial. La memoria en cambio, pese a ser el recurso más utilizado según el análisis previo, por las características del *Map server* se espera que aumente de forma constante. En cualquier caso, se dispone de un script que una vez al día monitoriza el consumo de recursos y envía un correo a los administradores en caso de que se alcancen ciertos umbrales predefinidos, por lo que el servidor nunca excederá su capacidad máxima por un descuido.

Por tanto, en caso de tener que aumentar alguna característica de la máquina, estas serían o la memoria o la CPU. En el primer caso el mantenimiento conllevaría parar el servicio durante unos minutos, el tiempo que los encargados de mantenimiento de la empresa de hosting tardaran en ampliarla, y en el segundo posiblemente se alargara más, pero no debería alargarse más de una hora.

### 8.1.3 Gestión de la continuidad

BidaiRO está disponible las 24 horas del día durante todo el año. En todo el tiempo que lleva abierto, sólo en contadas ocasiones ha dejado de prestar servicio por algún fallo del propio servidor, pero para minimizar el impacto en caso de que ocurriera se ha diseñado un script que monitoriza los servidores principales: *Login server*, *Char server*, *Map server*, *Anticheat server* y *MySQL*. En caso de que alguno de estos servidores deje de funcionar, los administradores recibirán un correo electrónico y un mensaje de texto al teléfono móvil, de forma que pueda restablecerse el servicio lo antes posible.

Para los casos en los que el problema esté fuera del alcance de la administración (incidencias del host, saturación de nodos, etc.), los pasos a seguir consisten en informar a los usuarios por el canal informativo más adecuado según la situación y observar la evolución del problema. Generalmente por problemas externos no es necesario interrumpir el servicio, aunque la experiencia de juego pueda verse afectada.

### 8.1.4 Tiempo de incidente

Se van a clasificar los incidentes en dos tipos: de **software** y de **hardware**. El primer grupo suele ser el más rápido de resolver, ya que no es necesario ponerse en contacto con la empresa de hosting. En las tablas siguientes se muestran algunos tipos de incidentes que pueden ocurrir y la forma de resolverlos, así como el tiempo que se puede tardar, basado en la experiencia del servidor.

Incidente	Tiempo de respuesta	Tipo
<b>Caída del servidor eAthena</b>	De 1 a 6 horas	Software
<b>Resolución</b>		
Generalmente son fallos que se dan en ocasiones puntuales. Es posible restaurar el servicio mientras se busca y arregla el problema. Si es persistente, puede restaurarse una copia de seguridad que no tenga el fallo para no interrumpir el servicio.		

Incidente	Tiempo de respuesta	Tipo
<b>Caída del servidor Apache</b>	De 1 a 2 horas	Software
Resolución		
El servidor Apache es extremadamente estable, por lo que una caída probablemente responda a un fallo en la configuración. Revisarla junto a los logs sería el primer paso a seguir.		

Incidente	Tiempo de respuesta	Tipo
<b>Caída del servidor MySQL</b>	De 1 a 3 horas	Software
Resolución		
Las posibles causas serían una mala configuración, la corrupción de alguna tabla importante o una consulta extremadamente pesada, aunque en este caso el servidor no se llega a caer, sino que se bloquea. Los pasos a seguir son revisar la configuración y los logs, donde también se obtendrán pistas del origen del problema. En caso de tablas corruptas siempre es bueno comprobar la integridad del disco duro y la memoria.		

Incidente	Tiempo de respuesta	Tipo
<b>Corrupción en la base de datos</b>	De 15 minutos a 4 horas	Software
Resolución		
Este incidente es relativamente común, puede ocurrir cada pocos meses aunque con un impacto leve y fácilmente identificable. El primer paso es restaurar las tablas corruptas, ya sea mediante una copia de seguridad o manualmente en caso de afectar sólo a algunos registros (suele ocurrir en algunas tablas de eAthena). Después es recomendable revisar los logs y la integridad del disco duro y la memoria para asegurar que el problema no viene de un incidente mayor. El tiempo de respuesta puede alargarse dependiendo de si hay que realizar una intervención manual y de su complejidad.		



Incidente	Tiempo de respuesta	Tipo
<b>Fallo en el disco duro</b>	De 2 a 4 días	Hardware
Resolución		
<p>Si se detecta a tiempo puede llegar a resolverse sin ningún impacto en el servidor, más que la parada en el servicio que realizarán los técnicos al sustituir el disco dañado. Si no, puede haber pérdida de datos y ser necesario restaurar backups, lo que alarga el tiempo de respuesta. Tener un RAID<sup>[19]</sup> bien configurado es una garantía para no perder los datos y poder seguir prestando servicio aun teniendo un disco estropeado. La resolución consiste en pedir al servicio técnico del host la sustitución del disco, y dependiendo del impacto que pueda tener, cerrar el servidor hasta que esté arreglado.</p>		

Incidente	Tiempo de respuesta	Tipo
<b>Fallo de memoria</b>	De 2 a 4 días	Hardware
Resolución		
<p>Es un incidente que puede ser complicado de detectar e impactar muy negativamente en el funcionamiento del servidor. La resolución consiste en pedir al servicio técnico del host la sustitución de la memoria y cerrar el servidor hasta que la cambien para evitar problemas mayores.</p>		

Incidente	Tiempo de respuesta	Tipo
<b>Cambio de host</b>	De 2 a 3 semanas	Hardware
Resolución		
<p>Un cambio de host siempre lleva cierto tiempo, ya que es necesario mover todo lo que había en el viejo al nuevo, configurarlo, reinstalar aplicaciones, redirigir dominios, etc. Normalmente al cambiar de máquina surge algún problema de incompatibilidad o es necesario cambiar el funcionamiento de algo. En el peor de los casos es posible que no se dispongan copias de seguridad del host anterior, lo que retrasaría aún más la migración.</p>		

### **8.1.5 Disponibilidad y tiempo de respuesta**

El servidor no tiene un horario de atención al usuario establecido, pero generalmente comprende de **7:30 a 23:00** los días laborables y de **11:00 a 23:00** los festivos. Durante esas franjas horarias se comprueba el foro y el correo al menos una vez cada hora, ya que son los canales que más utilizan los usuarios para comunicarse con la administración.

Las sugerencias, dudas o problemas de bajo impacto suelen resolverse al momento, mientras que intervenciones que requieran más tiempo suelen empezar a resolverse a partir de las **18:00** los días laborales y durante toda la franja de atención al usuario los festivos. Generalmente no se tarda mucho en resolverlas, pero si se alargan o se deben encolar por existir un volumen de trabajo alto, siempre se mantiene informados a los usuarios de los progresos realizados.

## **9 Conclusiones**

## **9.1 Conclusiones**

Tras más de dos años y medio con el servidor funcionando, se puede ver que la mayoría de los objetivos iniciales se han cumplido con creces. Incluso la organización de eventos, el punto donde siempre se flaqueó, se ha podido mejorar con la inclusión de los eventos automáticos, aunque siempre sería deseable organizar alguno presencial si el tiempo lo permitiera.

Sin embargo hay dos objetivos que no se han llevado a cabo: el correspondiente a la implementación de razas y al portal informativo. En el primer caso, generar los aspectos visuales de todas las razas, es algo muy laborioso, ya que hay que diseñar todos los puntos de vista del personaje multiplicados por el número de razas que se quisieran añadir. Al ser algo poco prioritario, ya que sólo cumpliría una función decorativa, nunca se ha llegado a implementar por existir cosas más importantes y urgentes.

El caso del portal informativo es similar, siempre ha tenido una prioridad baja al existir ya otros servicios similares. Además, aunque hay bastante información que puede obtenerse con una simple consulta a la base de datos, para mostrar otros datos es necesario realizar pasos más complejos ya que parte de la información está en ficheros de texto y es necesario tratarlos adecuadamente para relacionarlos entre sí y mostrar el contenido de forma adecuada (no es lo mismo mostrar una línea de texto que dibujar un mapa).

El servidor BidaiRO se trata de un proyecto muy completo, ya que para su realización ha sido necesario tocar varios puntos de diversa índole: contratación del dominio y el servidor dedicado, configuración y mantenimiento de la máquina, gestión de distintos tipos de servidores, gestión del equipo, diseño de la web, programación tanto para nuevas implementaciones propias como modificación de otras ya existentes, procesos de análisis, búsqueda de información, gestión del foro y trato con el usuario, resolución de incidencias, etc. Gracias a esto ha sido posible adquirir experiencia o afianzarse en una gran variedad de campos, algo que sin duda resultará útil en otros proyectos futuros.

Un punto importante a tener en cuenta es el trabajo en equipo que ha caracterizado la creación del servidor, sobre todo en su primer año. BidaiRO nunca habría nacido de no haber estado compuesto por un grupo de personas motivadas y con buenas aptitudes para, como mínimo, aprender nuevas cosas y mejorar en las que ya se conocían. Tras el primer paso de apertura al público, ha sido esencial la buena coordinación del equipo y la correcta asignación de prioridades, fechas y tareas a cada miembro según sus habilidades y conocimientos adquiridos a lo largo del tiempo. Un conocimiento global de las tareas asignadas a cada uno y sus avances y problemas a la hora de realizarlas permitió que la toma de decisiones para la gestión del proyecto fuera adecuada y eficaz en todo momento.

La progresiva salida del proyecto de varios de los miembros iniciales, aunque dificultó su desarrollo debido a la acumulación de tareas e inicialmente lo ralentizó debido al proceso de traspaso de conocimiento, tuvo como punto a favor la profundización en todas las áreas que componen el proyecto, ya que pese a haber tenido siempre un mínimo de conocimiento de lo que hacía cada uno y la forma en que se llevaba a cabo, en muchos casos esta información era más abstracta que en otros. La obligación de tomar el testigo en todos estos temas implicó el aprendizaje de los mismos y la consiguiente adquisición de todo el conocimiento y habilidades necesarias para poder realizar correctamente esas tareas, algo que siempre es positivo.

Este proyecto puede suponer el pistoletazo de salida para proyectos más grandes, una vez superada la fase inicial de habilitar toda la estructura y abrirla al público, algo que siempre crea cierta inquietud e inseguridad en el momento de comenzar. El que todo esto se haya podido realizar partiendo de una simple afición por el Ragnarok Online a manos de un equipo sin experiencia y se haya podido mantener pese a los problemas encontrados es alentador, y sin duda supondrá una gran ayuda, tanto psicológica como técnica, para la realización de nuevos proyectos con fines más prácticos.

## **10 Líneas futuras**

## **10.1      *Líneas futuras***

Un proyecto de este tipo puede expandirse de muchas maneras, ya sea añadiendo funcionalidades al propio juego, como se ha ido haciendo hasta ahora, o aprovechando la amplia comunidad de usuarios que tiene (no solo en el propio servidor, sino también fuera de él). Constantemente se han ido añadiendo actualizaciones, ya sea para mejorar o modificar las características del Ragnarok original o para incluir implementaciones propias. Sin embargo hay algunas tareas que requerirían de una planificación especial.

### **10.1.1      Servidor PK**

El Ragnarok Online no tiene PK, a diferencia de otros juegos como el World of Warcraft, donde hay dos facciones y es posible combatir si se entra en el territorio del enemigo. El PvP del RO, a diferencia de lo que se puede considerar PK tradicional, es pactado: dos o más personas entran en una sala específicamente diseñada para combatir, o luchan por un castillo en una WoE o eventos similares. Esto quiere decir que para hacer un servidor PK es necesario cambiar muchas cosas de la mecánica del juego, si se quiere hacer equilibrado.

En un principio, antes de la apertura del servidor, se barajó la idea de hacerlo PK. Se descartó por no tener ninguna experiencia en la gestión de servidores de este tipo, con la idea de comenzar con algo sencillo y tal vez, más adelante, implementar la PK. Se tuvo en cuenta la posibilidad de implementar una PK opcional a partir de nivel 90 renacido, de forma que quien quisiera pudiera optar por este camino y sólo pudiera avanzar de nivel llegando a un cierto número de puntos PK conseguidos al derrotar rivales. Estos puntos se perderían al morir a manos de otro jugador, de forma que sería necesario ser hábil para progresar. Obviamente quien no eligiera este camino no podría ser atacado por otros jugadores, ni atacarlos. Sin embargo se consideró que la mejor solución sería no mezclar la PK con el servidor normal y en vez de eso implementar todo el sistema en un nuevo servidor exclusivamente PK desde la creación del PJ.

Para crear un servidor PK hay que tener en cuenta varias cosas, entre ellas:

- **Rates:** En un servidor PK se muere más que en uno normal, y de forma más inesperada. Esto implica una pérdida de experiencia mayor y más dificultad para subir de nivel, por lo que los rates deberían ser lo suficientemente altos como para poder subir sin demasiado esfuerzo y lo suficientemente bajos como para no subir muy fácilmente.
- **Equipo:** Antiguamente, la mayoría de los MUDs era PK y al morir el atacante podía saquear el cuerpo, robando todo el equipo. Hay que recordar que los MUDs fueron los precursores de los MMORPGS, y que aunque actualmente este tipo de juegos comparten la misma base, las características que había por aquel entonces eran mucho más sencillas. Los monstruos no tenían una probabilidad de soltar equipo al morir, siempre se obtenía el equipo, de forma que perderlo no era tan importante como en un MMORPG actual, en el que para obtener ciertas piezas de equipo pueden emplearse días e incluso meses. Esto plantea la duda de cómo implementar el saqueo y el robo de objetos, y también tiene mucha relación con los rates (de equipo, no de experiencia) mencionados anteriormente. Posibles soluciones son no permitir el saqueo, hacerlo con una probabilidad baja, vender en tiendas equipo considerado básico, facilitar la obtención de equipo, permitir marcarlo para que vuelva a su propietario, etc.
- **Facciones:** Una característica muy interesante de un servidor PK son las facciones. A diferencia de los clanes, las facciones serían agrupaciones de individuos a gran escala, los buenos y los malos. Un posible servidor PK podría tener tres facciones, una buena, otra mala y otra de renegados, que serían los que van por libre. Los buenos sólo podrían atacar a los malos y a los renegados que les provocaran, los malos podrían atacar a todos, incluso entre



ellos, y los renegados podrían atacar a otras facciones y a la suya propia dependiendo de la zona en la que se encontraran sus principales asentamientos, dependiendo de la historia que tuvieran en cada región.

- **Mapa:** Esto tiene mucha relación con las facciones y el equipo. El mapa del Ragnarok tiene varias ciudades y zonas que podrían repartirse entre las distintas facciones, de forma que cada una tuviera sus propios territorios, de forma similar al World of Warcraft. Sin embargo esto puede suponer problemas a la hora de obtener equipo, ya que hay objetos básicos que sólo se pueden encontrar en ciertos mapas del juego, y la facción que los controlara tendría ventaja sobre las demás. Esto podría solucionarse duplicando mapas o creando instancias, aunque la mejor opción sería rehacer por completo el mundo; llevaría mucho trabajo y probablemente surgirían ciertos problemas, pero podría obtenerse un resultado muy bueno.
- **Zonas seguras:** Es necesario decidir si va a haber zonas seguras o no. En un servidor PK total, lo más seguro que se puede encontrar son las ciudades de cada facción, que estarían protegidas por guardias controlados por el propio juego. Esto no quiere decir que una ciudad sea invulnerable, sino que si es lo suficientemente grande y el enemigo no es lo suficientemente poderoso, sus habitantes se podrán encontrar a salvo. Esto da mucho juego porque es posible organizar incursiones. Otra opción es definir zonas en donde no se pueda pelear, que podrían ser las propias ciudades o ciertos mapas.
- **Combate:** Sería necesario ajustar el combate entre jugadores, ya que el Ragnarok no está pensado para eso. Por ejemplo, el daño realizado por ciertos jobs es muy alto y posiblemente desequilibre mucho el juego: un asesino al acecho puede eliminar de un solo golpe a la mayoría de los jobs si no saben que van a ser atacados.

Para ello puede seguirse el método de las WoEs, donde el daño se reduce aproximadamente un 50% (dependiendo de si es un ataque cuerpo a cuerpo, de rango o de alguna habilidad), pero es posible que hubiera que hacer algunos ajustes en ataques específicos. También hay que tener en cuenta que en el servidor oficial existen objetos que te llevan automáticamente a una ciudad estés donde estés. Esto puede perjudicar mucho a los combates porque cualquiera que quiera huir sólo tiene que pulsar una tecla. Para evitar esto se podría implementar un temporizador que inhabilitara estos medios de transporte durante unos minutos después de haber recibido un golpe o visualizado un enemigo.

Como se puede ver, son muchos aspectos a tener en cuenta y es necesario trabajar en un servidor diseñado específicamente para este sistema de juego, ya que el Ragnarok actual es totalmente incompatible con esta filosofía.

### **10.1.2 Razas**

Se dice que en el Ragnarok Online 2 se van a incluir tres razas, pero el RO actual no tiene ninguna. Esto es una posible implementación que, según cómo se realizara, podría hacerse en el servidor actual o en uno nuevo.

Algo compatible con cualquier tipo de servidor serían las razas como aspecto visual de los personajes. Es lógico que una raza tenga bonificaciones y penalizaciones en algunas características, habilidades propias, resistencias y vulnerabilidades a elementos y ataques, e incluso lenguajes independientes, pero esto se saldría del funcionamiento oficial del juego, cosa que en BidaiRO no se quiere. Por tanto se pueden implementar razas que cambien el aspecto del personaje sin influir en nada más. Esto podría hacerse al momento de empezar a jugar, tras pasar la fase de entrenamiento donde se suben los primeros niveles. Dependiendo de la raza elegida, el personaje se vería de una u otra forma, por ejemplo pequeño para enanos, grande para gigantes, con orejas puntiagudas para elfos, de color verde para trolls, etc.

Aparte del aspecto visual, lo interesante es también implementar las bonificaciones y penalizaciones antes comentadas. Esto puede ser muy interesante en el caso del servidor PK, en donde no se seguirían las reglas del servidor oficial de Gravity. Además, cada raza tendría asociada una facción, y en algunos casos podría escogerse (un humano podría ser bueno, malo o renegado, mientras que un elfo sólo podrá ser bueno, por ejemplo).

A continuación se muestra la *Tabla 14: Lista de razas* con las posibles razas que se barajaron antes de abrir el servidor, y unos ejemplos visuales en la *Imagen 30*:

*Ejemplo de elfo oscuro, Imagen 31: Ejemplo de no-muerto e Imagen 32: Ejemplo de orco.*

Humanos	Elfos	Elfos oscuros	Enanos	Gigantes	No-muertos
Pitufos	Orcos	Espíritus	Trolls	Hombres-pep	Hadas
Gnomos	Draconianos	Elementales	Genios	Ángeles	Demonios

**Tabla 14: Lista de razas**



**Imagen 30: Ejemplo de elfo oscuro**



Imagen 31: Ejemplo de no-muerto



Imagen 32: Ejemplo de orco

### 10.1.3 Servidores múltiples

Al igual que Gravity tiene varios servidores oficiales distribuidos por el mundo y estos tienen más o menos características distintas entre sí, no es raro que un servidor privado ponga a disposición de los usuarios varios servidores de distintas características. Lo más normal en este caso es abrir servidores parecidos pero con varios rates.

En el caso de BidaiRO, sería posible abrir un servidor de rates medios y otro de rates altos. Sin duda este último sería el que más gente atrajera, ya que son los preferidos de los jugadores. Además, al ser un servidor conocido mucha gente jugaría gracias a la fama que tiene en España y algunos países de América del sur. El emulador eAthena proporciona las funcionalidades necesarias para abrir más de un servidor aprovechando los recursos de la máquina, por lo que esa parte sería sencilla.

Sin embargo habría que realizar muchos cambios importantes. Para empezar sería necesario o bien rediseñar la web para que incluyera los tres servidores, o

preferiblemente crear una para cada servidor, con algunas páginas en común que sirvan de enlace. Esto es sencillo en el caso del foro, donde basta con crear nuevos subforos para cada servidor y reestructurar los ya existentes, pero para el resto del portal requiere una etapa de análisis y de diseño importante. Además, la comunicación que hay a través de la web con el servidor y los distintos componentes que componen el proyecto están pensadas para interactuar con un solo servidor, y es probable que hubiera que realizar modificaciones ahí también.

En el caso de los nuevos servidores, sería necesario cambiar la política de respetar al máximo el original. Cuando a un servidor se le suben los rates, es necesario añadirle muchas personalizaciones como equipo nuevo, aumento del máximo de niveles, del máximo de stats, eventos y otras tareas similares indispensables para mantener el interés en los jugadores. Cuando es posible alcanzar el nivel máximo en unos pocos días, son necesarios más alicientes para seguir jugando.

Obviamente, no sólo se podrían abrir servidores con rates más altos: el servidor PK sería un candidato perfecto para acompañar al actual.

#### **10.1.4 Portal informativo**

Existen varias páginas web con información y ayuda sobre el Ragnarok que son consultadas asiduamente por los jugadores. Estas páginas ofrecen foros, guías, anuncios y novedades del servidor oficial, y bases de datos donde consultar objetos, monstruos, habilidades y cualquier cosa relativo al juego. Tampoco es raro que tengan su propio top de servidores y les ofrezcan la posibilidad de anunciarse en distintos lugares de sus portales previo pago.

Posiblemente una de las más conocidas y utilizadas sea **RateMyServer**<sup>[182]</sup>, cuya sección de búsqueda se muestra en la *Imagen 33: Base de datos de RateMyServer.net*.

Home Log off Forum

Item - Item Type - Monster Skill

SERVER LISTING

Low Rate  
Mid Rate  
High Rate  
Super High Rate

List All Search

DATABASES

- Item Search
  - Weapons
  - Armors
  - Cards
  - Ammunitions
  - Others
- Monster Search
- Skill Database
- Quest Database
- RO Writings
- Map Database
- Creation Database
- Misc Database
- NPC Search
- Shop Search

..Item Database..

El juego de simulación histórico-económico

Europe 1400

JUEGA GRATIS ¡YA!

Current Item Drop Rate is set to 1x

**Bobbin Of Goddess** Item ID# 7907 (Bobbin\_Of\_Goddess) Spw

Type	Miscellaneous	Class	n/a	Buy	n/a	Sell	n/a	Weight	0
Description	Not Available								
Item Script	{},{,}								
Dropped By	No Result								

**Emblem of the Sun God** Item ID# 7086 (Symbol\_Of\_Sun) Spw

Type	Miscellaneous	Class	n/a	Buy	1z	Sell	0z	Weight	50
Description	A golden, shining bead that is a symbol of the god of the sun.								
Item Script	{},{,}								
Dropped By	Treasure Chest (0.8%)								

Is a quest ingredient of:  
Hat of Sun God, Sunday Hat, Mage Hat & Magician Hat

**God Anvil** Item ID# 7806 (Black\_Anvil) Spw

Type	Miscellaneous	Class	n/a	Buy	n/a	Sell	n/a	Weight	1
Description	" Holffar " Received a God's Anvil.								
Item Script	{},{,}								
Dropped By	No Result								

Imagen 33: Base de datos de RateMyServer.net

Un servicio rápido de implementar sería el de búsquedas en la base de datos. Mediante un interfaz sencillo se podrían realizar consultas desde la web a la base de datos para obtener cierta información. Esto sería un buen comienzo, pero insuficiente para competir con las páginas que hay actualmente. La información relativa a objetos y monstruos se obtiene directamente de la base de datos, pero las habilidades, mapas, tiempo que tarda en renacer un monstruo y otros muchos datos están almacenados en ficheros de texto que el servidor lee y procesa una única vez cada vez que arranca. Sería necesario crear una base de datos propia donde replicar las tablas actuales que se podrían consultar (por seguridad y para evitar sobrecargas en el juego es mejor separarlas aunque los datos sean los mismos), y después crear nuevas tablas con toda la información necesaria de estos ficheros. Mediante un script se procesaría la información de estos ficheros y se insertaría de forma automática en las tablas para poder ser consultada a través de la web.

Con todo esto creado también sería muy interesante eliminar la dependencia del emulador de estos ficheros de texto y leerlo todo de tablas, pero ya sería un tema aparte.

## **11 Planificación y costes**



## **11.1 Planificación y costes**

Este proyecto puede dividirse en dos fases: una de creación y puesta en marcha, donde partiendo desde cero se obtiene un servidor completamente funcional con todo lo básico para operar, y otra de actualizaciones y mantenimiento, que se centra en arreglar fallos, dar soporte a los usuarios, mantener actualizado el servidor e implementar cualquier cosa que pueda ser interesante tanto para el usuario como para la administración.

Obviamente la primera fase es la que puede concretarse más, ya que es la que ha requerido una planificación mayor. En la segunda fase se normalmente se planifican implementaciones complejas, resolución de problemas graves y actualizaciones periódicas, pero generalmente las implementaciones sencillas y el soporte al usuario se lleva a cabo según va siendo necesario. Por esto se va a detallar toda la primera fase de planificación y para la segunda se hará una estimación mensual, que podrá variar dependiendo de la carga de trabajo que se tenga en cada momento a raíz de las implementaciones pendientes, fallos que surjan, incidencias de los usuarios y recursos humanos disponibles.

El coste requerido para poner en marcha el proyecto y para mantenerlo también es diferente.

### **11.1.1 Fase 1: Creación del proyecto**

Lo primero es tener claro qué es lo que se quiere ofrecer, cómo se va a hacer y quién se ocupará de cada cosa. Y por supuesto cómo llamarlo. Partiendo de que no se tiene experiencia, se prefiere comenzar con un servidor sencillo, que funcione sin fallos pero con alguna implementación personalizada que haga ver a los nuevos usuarios que el servidor no es uno cualquiera que se ha abierto sin más, sino que se tiene la capacidad de innovar e implementar ideas propias sin necesidad de depender exclusivamente de lo que haga el equipo de eAthena.

A nivel del emulador, se empieza configurándolo al propio gusto. eAthena tiene múltiples opciones de configuración con las cuales se pueden obtener servidores de

características muy diferentes usando exactamente el mismo código. Se enumeran todas las opciones disponibles y se decide qué valores se les asignará. Viendo esto ya se pueden empezar a pensar en algunas personalizaciones para realizar antes de la apertura o más adelante, cuando el servidor funcione sin problemas y empiece a tener usuarios. En este punto se pueden empezar a hacer pruebas sencillas instalando el emulador en casa, y se puede avanzar algo en las personalizaciones más sencillas.

Todo emulador necesita un cliente, por lo que es necesario documentarse sobre la creación del mismo y empezar a generar uno sencillo, que poco a poco se irá completando y configurando a la medida del servidor. En el Ragnarok es importante el aspecto visual de los personajes y los objetos, por lo que es necesario contar con un paquete de tintes para la ropa, estilos de pelo, objetos coloreados o personalizados, etc. Existen distintos paquetes disponibles de forma gratuita, por lo que hay que escoger cuál utilizar y conseguir que funcione en el cliente. Más adelante se pueden añadir personalizaciones o modificar lo que se está usando, incluso con el servidor abierto.

Con la creación del cliente ya se pueden empezar a perfilar algunos aspectos básicos para evitar que los usuarios hagan trampas. Son sistemas rudimentarios pero que en muchos casos sirven para desalentar a los usuarios menos hábiles. Se es consciente de que hasta que no se implementen sistemas más seguros hay que vigilar a los usuarios de forma presencial para complementar los sistemas que se vayan a implementar. Estos sistemas pueden requerir que se modifiquen algunos aspectos de la configuración del servidor.

Para ofrecer todo esto es necesario un portal web donde se presente el servidor, se detallen sus características y personalizaciones y a ser posible que tenga algunas secciones básicas de descarga del cliente, normas y ayuda. Se decide permitir la creación de cuentas de juego a través de la web e implementar un formulario para que el usuario pueda contactar con la administración de forma rápida y sencilla. También es necesario instalar un foro, para el cual hay que decidir los subforos que se añadirán y su estructura, y un panel de control, ambas cosas imprescindibles para un servidor de Ragnarok.

Dado que la web es lo primero que ve el usuario, se considera imprescindible dedicarle todo el tiempo necesario para que proporcione una buena imagen del servidor. La primera versión de la web disponible en 2008 era muy similar a la actual, la estructura era prácticamente la misma y los cambios que se han ido haciendo han consistido en la inclusión de algunas secciones, servicios como el bugtracker y muchas personalizaciones en el panel de control para ofrecer más información sobre el juego a los jugadores.

Todo esto es necesario alojarlo en un host y disponer de un dominio propio. Es posible probarlo todo en un servidor Linux casero, pero antes de la apertura hay que disponer de una máquina donde subirlo todo. Contratar el dominio es sencillo, basta buscar uno con un buen precio, protección de privacidad Whois<sup>[120]</sup> y buenas referencias; en general todos tienen características similares. Para el host, se hace una estimación de la potencia que deberá tener la máquina para correr todos los servidores (eAthena, Apache, MySQL, etc.) y el ancho de banda del que dispondrá. Se pone un máximo estimando de unos 100 usuarios, cifra que se ha llegado a alcanzar alguna vez pero que normalmente es menor, y se buscan hosts cuyas características sean ligeramente superiores a esa estimación que tengan una buena relación calidad-precio. Para ello se busca información en foros y se usan referencias de otros servidores que se han probado anteriormente. Se decide que el ancho de banda mensual deberá ser ilimitado, el sistema operativo será alguna distribución de Linux y que la empresa de hosting deberá proporcionarnos acceso total a la máquina.

Las características de la máquina actual no difieren mucho de las que ofrecía el primer host contratado para la apertura (son ligeramente mejores por un precio similar, como suele ocurrir con el paso del tiempo), y se muestran en la *Tabla 15*:

*Características del host (máquina y red).*

<b>CPU</b>	Intel Core2Duo 2x 2.33+ GHz
<b>Memoria RAM</b>	2 GB
<b>Disco duro</b>	2 x 500 GM (RAID)
<b>Sistema operativo</b>	GNU/Linux Ubuntu 64 bits
<b>Conexión</b>	100 Mbps
<b>Tráfico mensual</b>	Ilimitado

Tabla 15: Características del host (máquina y red)

El consumo de memoria RAM suele ser un 60% del total, la swap<sup>[121]</sup> un 10%, la CPU está entre un 1% y 3% y el disco duro utilizado es del 40% en la partición<sup>[122]</sup> /home<sup>[123]</sup> (de 204 gigas) y del 10% en /<sup>[124]</sup>. El tráfico de la red suele ser de unos 60 kb/s de entrada y 190 kb/s de salida. Todos estos datos demuestran que el servidor es más que capaz de soportar la carga actual, y que si en un futuro aumentaran drásticamente los usuarios, con añadir más memoria sería suficiente si resultara necesario, por tanto la planificación en cuanto a requerimientos técnicos ha sido muy buena.

Con el host contratado es necesario configurar la máquina: crear usuarios, permisos y árboles de directorios, planificando dónde irá cada cosa y qué usuarios podrán utilizarlas. Se crean, además de los usuarios de personas físicas, usuarios para el resto de servidores, incluyendo el de eAthena de producción y de desarrollo. Se procede a optimizar las características de la máquina, sobre todo en lo relativo al uso de la red, a asegurarla contra posibles intrusos y a configurar los logs, y posteriormente se instalan todos los servicios necesarios y los servidores, se configuran (en el caso de Apache y MySQL este paso se realiza directamente en la máquina del host, ya que es donde mejor se puede ver cómo afecta la configuración) y se optimizan. Estos últimos pasos requieren documentarse bastante sobre el funcionamiento de cada componente.

Finalmente, cuando todo está funcionando, se procede a habilitar el portal web y a publicitar el servidor en foros y tops. Se debe publicar una presentación por parte de la administración y una fecha de apertura de la beta con rates más altos. Hasta entonces el juego estará cerrado a los usuarios y se irán ultimando los últimos detalles, pero se permiten la creación de cuentas. Una vez abierta la beta, los jugadores tendrán un

tiempo para probar el servidor y reportar los fallos que encuentren. Durante ese tiempo se corregirán los fallos y se ajustará lo que se considere necesario, y tras la finalización de la beta se borrará la base de datos del emulador para recomenzar de cero y se abrirá la versión oficial con los rates normales.

A continuación se muestra la **Error! Reference source not found.** y la **Error! Reference source not found.**, donde se puede observar la planificación del servidor desde el momento de su apertura hasta el final de la primera fase.

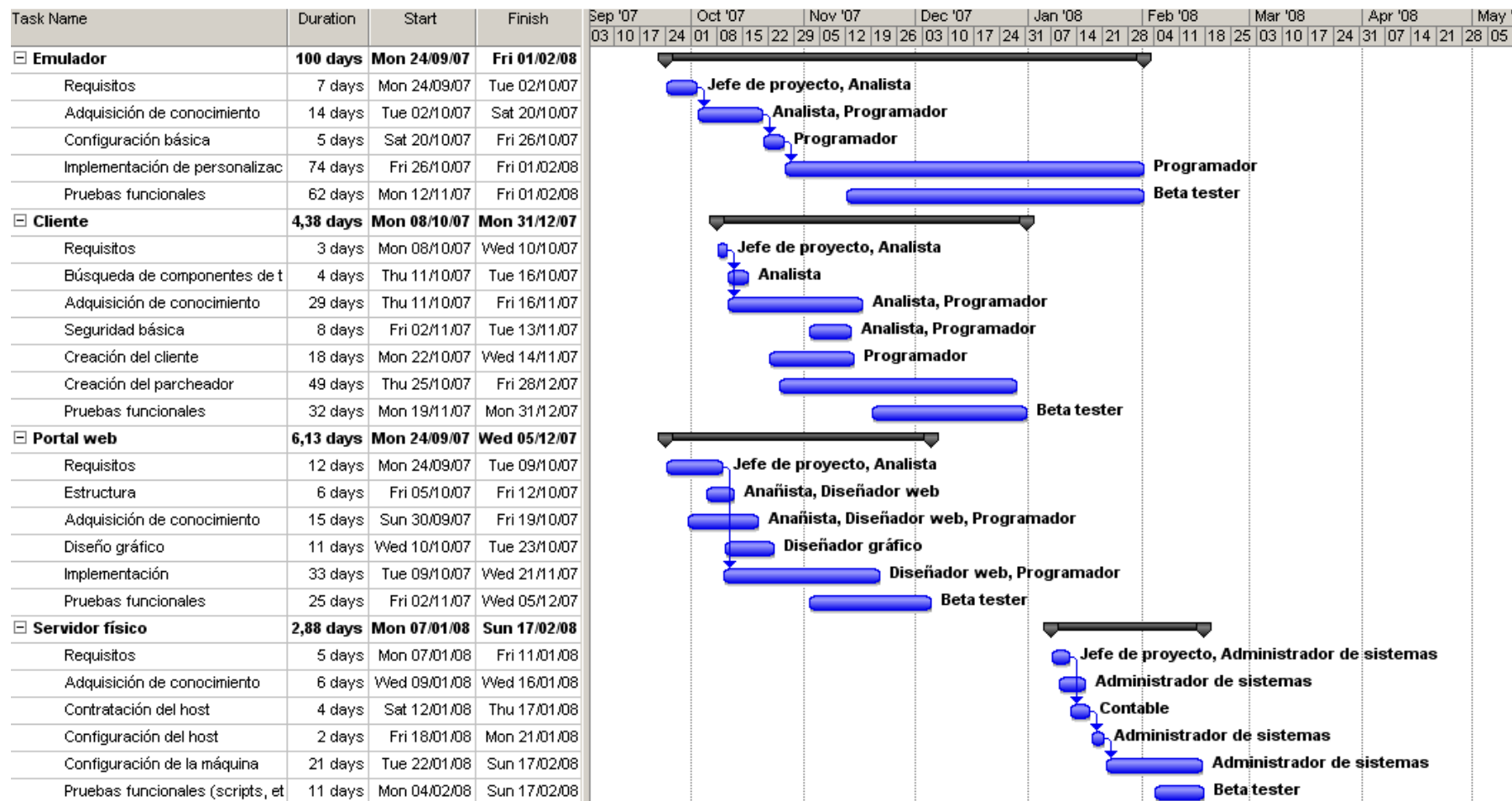


Imagen 34: Planificación inicial Fase 1 (1 de 2)

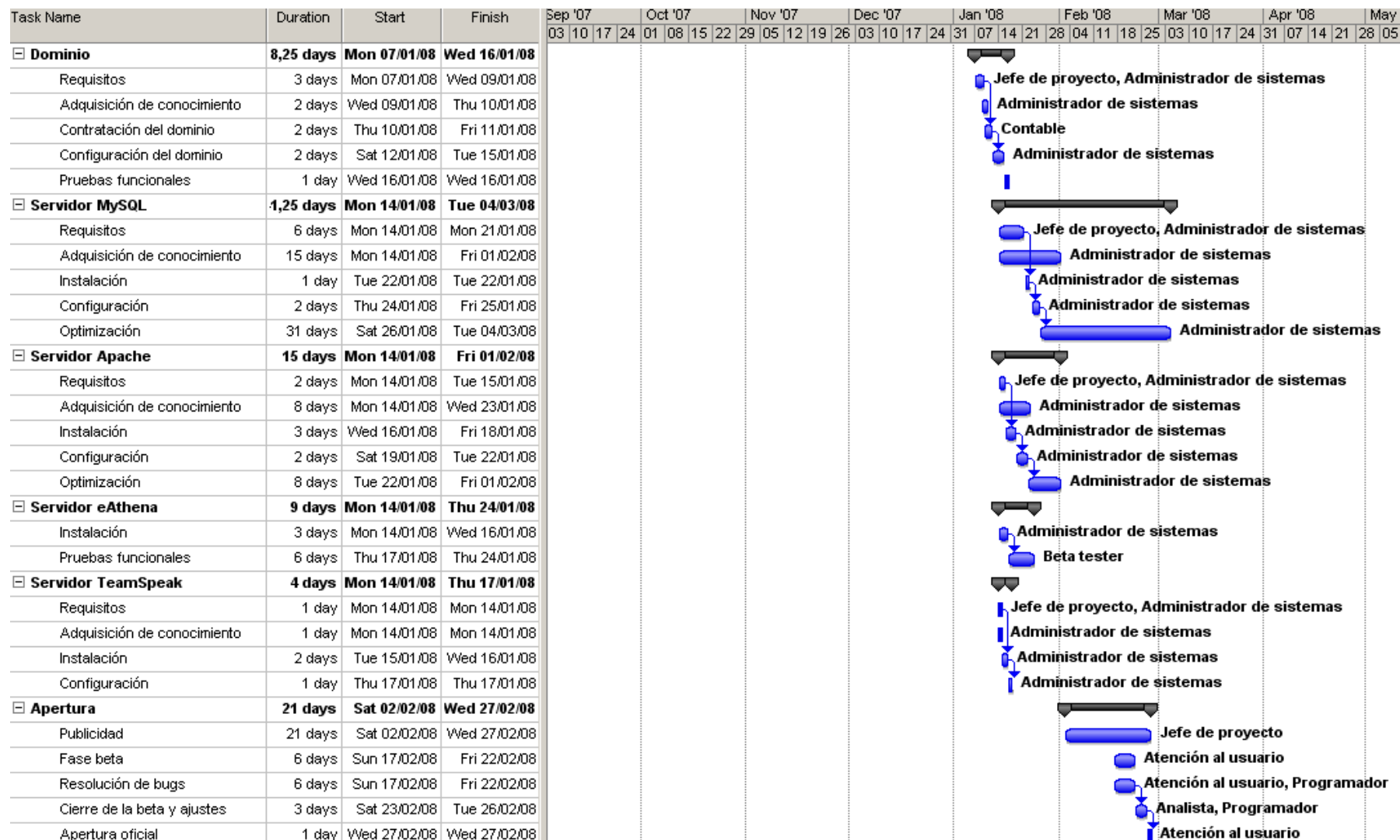


Imagen 35: Planificación inicial Fase 1 (2 de 2)

Pese a haberse mantenido las fechas de apertura, no toda la planificación se pudo llevar a cabo en las fechas previstas. A continuación se resumen las desviaciones producidas.

- **Emulador**

Se observa un retraso en las tareas de *configuración e implementación de personalizaciones* debido a la inexperiencia con eAthena. Esto provoca también retrasos en las *pruebas funcionales*.

- **Cliente**

Se observa un leve retraso en la *adquisición de conocimiento*, ya que es difícil encontrar información para algunos aspectos más concretos. Las tareas de *seguridad básica, creación del cliente y creación del parcheador* también se retrasan ligeramente, pero las *pruebas funcionales* se realizan antes de lo esperado.

- **Portal web**

Se observa un retraso en la *adquisición de conocimiento* para la parte más técnica del desarrollo, y un leve retraso en la tarea de *diseño gráfico*. La *implementación* por el contrario se realiza antes de lo previsto por resultar más sencilla de lo esperado.

- **Servidor físico**

Se lleva todo a cabo según la planificación original, incluyendo la tarea de *pruebas funcionales* que se finaliza antes de lo previsto.

- **Dominio**

Se lleva todo a cabo según la planificación original.

- **Servidor MySQL**

Se observa un retraso en la *adquisición de conocimiento*, mientras que la *instalación y la configuración* se adelantan. La tarea de *optimización*



lleva más tiempo de lo previsto pues es necesario que el servidor lleve un mínimo de tiempo funcionando para obtener datos más fiables.

- **Servidor Apache**

Se observa un leve retraso en la *adquisición de conocimiento* y en la *configuración*. Al igual que para el servidor MySQL, la *optimización* también se retrasa.

- **Servidor eAthena**

Se lleva todo a cabo según la planificación original.

- **Servidor TeamSpeak**

Se lleva todo a cabo según la planificación original.

- **Apertura**

Se lleva todo a cabo según la planificación original.

Estas desviaciones pueden verse de forma gráfica en la *Imagen 36: Planificación final Fase 1 (1 de 2)* y en la *Imagen 37: Planificación final Fase 1 (2 de 2)*, mostradas a continuación.

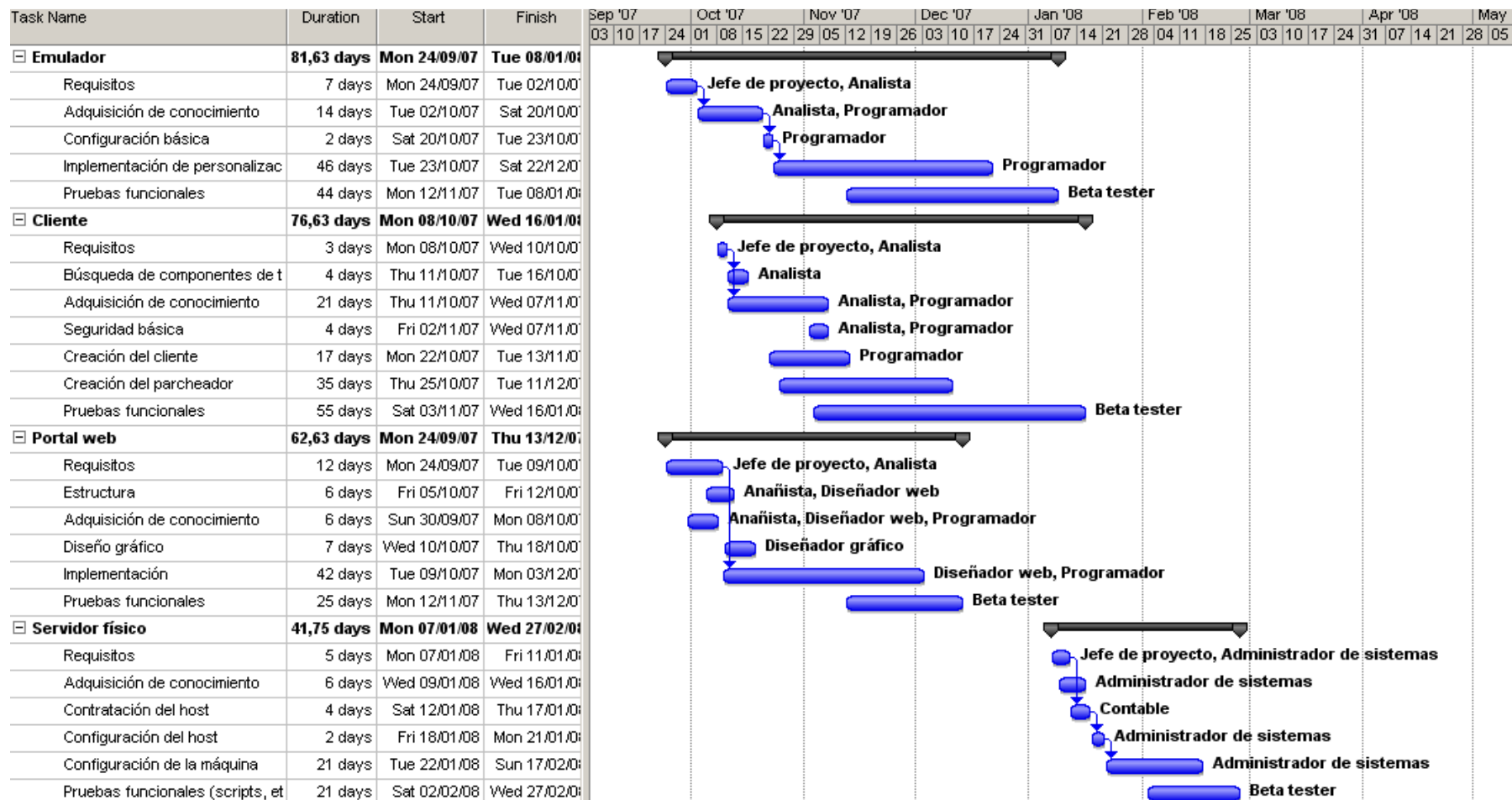


Imagen 36: Planificación final Fase 1 (1 de 2)

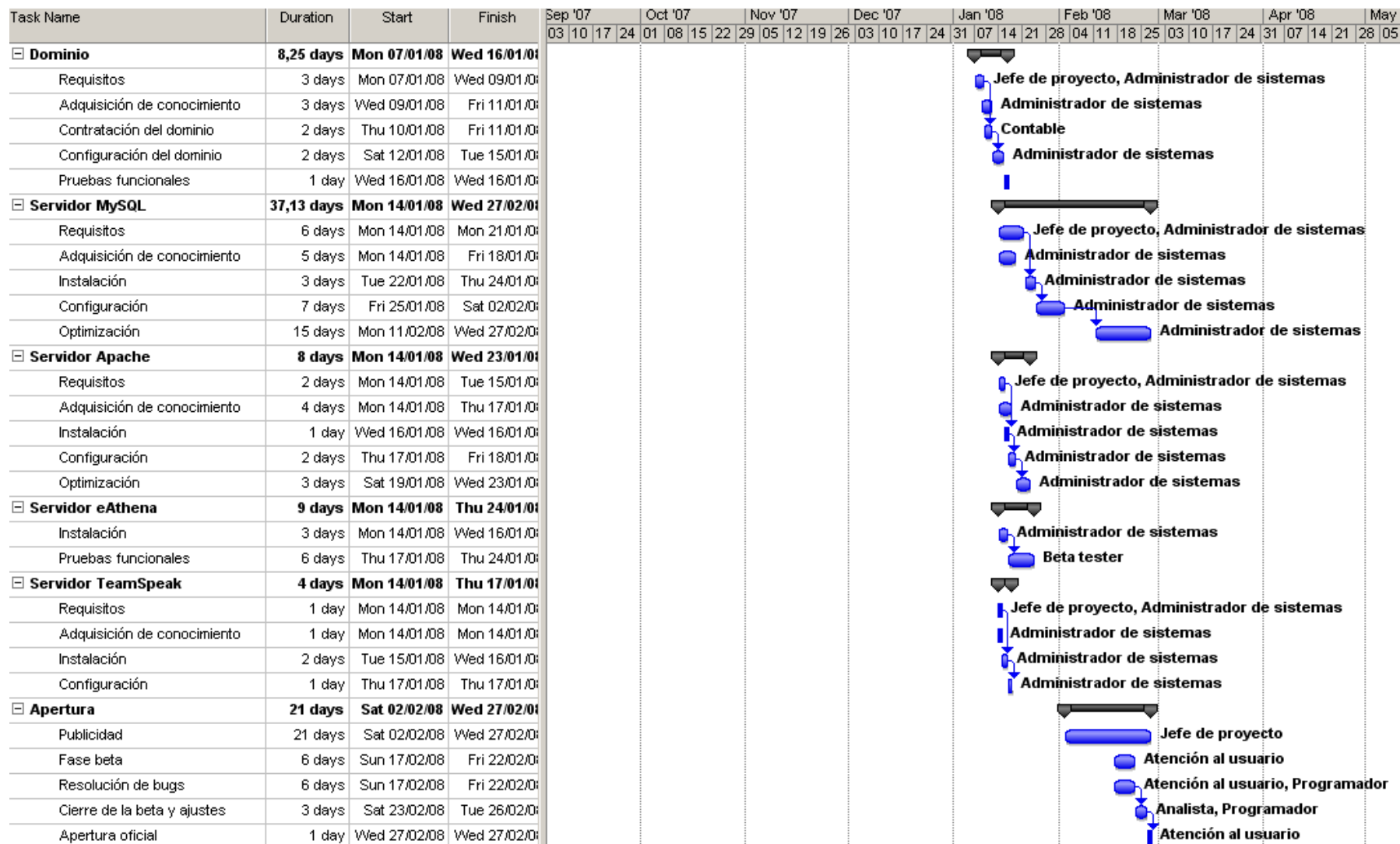


Imagen 37: Planificación final Fase 1 (2 de 2)

Teniendo esta información puede obtenerse el coste total de esta primera fase del proyecto, que se muestra a continuación en la *Tabla 16: Coste del servidor Fase 1 (1 de 2)* y la *Tabla 17: Coste del servidor Fase 1 (2 de 2)*.

Categoría	€/hora	Horas	Total €
Jefe de proyecto	17,27	480	8.289,60
Analista	10,97	720	7.898,40
Diseñador gráfico	7,67	72	552,24
Diseñador web	7,67	424	3.252,08
Administrador de sistemas	15,48	872	13.498,56
Programador	9,59	1.480	14.193,20
Beta tester	9,59	1.184	11.354,56
Atención al usuario	9,26	104	963,04
Contable	8,04	48	385,92
<b>Total</b>		<b>5.384</b>	<b>60.387,60</b>

Tabla 16: Coste del servidor Fase 1 (1 de 2)

Recurso	€/unidad	Unidades	Total €
Ordenador con Windows XP	669	6	4.014
Adobe Flash Catalyst CS5	529,82	1	529,82
Servidor dedicado	57,82	1	57,82
Dominio	19,95	1	19,95
Privacidad Whois	8,95	1	8,95
<b>Total</b>			<b>4.630,57</b>

Tabla 17: Coste del servidor Fase 1 (2 de 2)

Por tanto el coste total del proyecto en su primera fase es de **65.018,17€**.

### 11.1.2 Fase 2: Mantenimiento del proyecto

Una vez se ha abierto el servidor y está todo funcionando, es necesario mantener lo que ya se tiene (dar soporte al usuario y resolver bugs que se vayan encontrando), actualizar periódicamente el emulador para implementar las novedades del Ragnarok y así evitar acabar con un servidor desactualizado, e implementar personalizaciones y mejoras de lo que ya existe para ofrecer novedades de las que no dispongan otros servidores o que no vengan por defecto con el emulador.

La única planificación que se puede hacer de esto a priori es la actualización del emulador, que se hace mínimo una vez al mes aprovechando que hay que rotar el castillo de WoE y es necesario reiniciar el servidor, y el pago mensual del host. En cambio, la atención al usuario es algo constante pero no es algo que se pueda concretar, ya que hay días que hay que realizar muchas intervenciones y otros en los que no hay ninguna. Además la complejidad de las mismas puede ser muy variable. Incluso la carga de trabajo para actualizar el emulador puede variar dependiendo del volumen de las actualizaciones y los problemas que puedan surgir si interactúan con implementaciones propias del servidor.

Lo mismo ocurre con las implementaciones propias, puede haber algunas cosas que requieran de una planificación concreta pero generalmente esto no es necesario. Sí hay algunos eventos anuales que se planifican con antelación, y son los correspondientes al día de Navidad, Halloween<sup>[126]</sup> y el aniversario del servidor. Para estas fechas se implementan eventos especiales de uno o más días de duración y se realiza una planificación para tenerlos a tiempo y mejorar su implementación, ya que suelen tener una cierta complejidad.

A continuación se muestra la *Imagen 38: Diagrama de Gantt Fase 2*, la *Tabla 18: Coste del servidor Fase 2 (1 de 2)* y la *Tabla 19: Coste del servidor Fase 2 (2 de 2)* donde se observan los tiempos y costes aproximados que puede tener el mantenimiento mensual del servidor, pero como se ha dicho esto puede variar dependiendo de la carga de trabajo que se tenga en ese mes en concreto, por tanto lo único seguro es la actualización del emulador y los pagos del host y el dominio (este último se realiza anualmente, pero se mostrará su precio mensual para que quede reflejado).

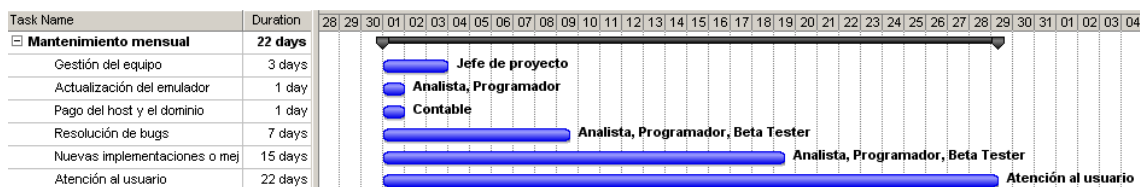


Imagen 38: Diagrama de Gantt Fase 2

Categoría	€/hora	Horas	Total €
Jefe de proyecto	17,27	24	414,48
Analista	10,97	36	394,92
Programador	9,59	184	1.849,2
Beta tester	9,59	54	541,08
Atención al usuario	9,26	176	1.098,24
Contable	8,04	8	89,6
<i>Total</i>		<b>482</b>	<b>4.785,9</b>

Tabla 18: Coste del servidor Fase 2 (1 de 2)

Recurso	€/unidad	Unidades	Total €
Servidor dedicado	57,82	1	57,82
Dominio	1,66	1	1,66
Privacidad Whois	0,76	1	0,76
<i>Total</i>			<b>60,24</b>

Tabla 19: Coste del servidor Fase 2 (2 de 2)

Por tanto el coste de mantenimiento mensual del proyecto es de aproximadamente **4.785,9€**.

En el *Anexo Digital A* se puede consultar una hoja de presupuesto más detallada.

## 11.2 *Plan de negocio*

Para recuperar la inversión realizada, es necesario crear un plan de negocio donde se obtengan los beneficios suficientes de diversas fuentes. Dado que un servidor privado se caracteriza por ser gratuito, no es viable cobrar a los jugadores una tasa mensual, pero sí es posible ofrecerlo de forma opcional a cambio de ventajas o vender ítems a precio reducido. Aunque esta no es la política de BidaiRO, es una de las mejores opciones para obtener beneficios. Además, pueden ofrecerse distintas tasas mensuales con distintos tipos de beneficio.

Aparte de esto, es posible obtener ingresos mediante publicidad. Para ello actualmente se está utilizando el servicio **ADSense** de Google, integrando en el foro los banners que ofrece, pero también sería posible añadir más banners en las distintas páginas de la web.

También es posible contratar servicios de mensajes de texto dinámicos en la página web principal. Estos servicios consisten en un formulario donde el usuario puede escribir un texto que aparecerá publicado en las páginas donde se habilite, pero para ello previamente es necesario enviar un **SMS**<sup>[125]</sup> a través del móvil. Hay que tener en cuenta que parte del coste del SMS se lo queda la empresa que oferta el servicio.

Otra opción alternativa, que es en realidad una mezcla de la anterior y la de venta de ítems en el juego, es habilitar el envío de SMS para recibir un código que posteriormente podrá utilizarse dentro del juego. Una vez usado, el jugador obtendría un objeto o paquete de objetos aleatorios de un solo uso, de forma que mezclando el azar con la caducidad de los objetos se asegurarían varios pagos de este tipo por muchos jugadores, sobre todo si los objetos entregados son útiles.

Para que estas ideas reporten beneficios deben cambiarse dos de los objetivos del servidor: las **donaciones no desequilibrantes** y la **mecánica fiel al oficial**. El primero para que aumenten las donaciones, que repercuten directamente en el beneficio mensual obtenido, y el segundo para poder implementar personalizaciones y formas de obtener objetos y ventajas de forma personalizada, previo pago. Además, es necesario

disponer de un número elevado de jugadores, ya que no todos estarán dispuestos a donar. Esto se consigue de forma indirecta a partir del cambio de objetivos, ya que son características que suelen atraer a un mayor número de usuarios.

Suponiendo que se dispongan de 1.000 jugadores, se realizará una estimación de los beneficios que podrían suponer las medidas explicadas anteriormente. En la *Tabla 20: Beneficios mensuales* y en la *Tabla 21: Beneficios diarios*, se muestra esta estimación.

Medida adoptada	Número de jugadores	Precio €	Total €
Tasa mensual plata	300	2	600
Tasa mensual oro	200	5	1.000
Tasa mensual platino	100	10	1.000
ADSense	-	40	40
<b>Total €</b>			<b>2.640</b>

Tabla 20: Beneficios mensuales

Medida adoptada	Número de jugadores	Precio €	Total €
Venta de ítems	20	2.5	50
Mensajes de texto	120	0.6	72
Venta vía SMS	80	0.6	48
<b>Total diario €</b>			<b>170</b>
<b>Total mensual €</b>			<b>5.270</b>

Tabla 21: Beneficios diarios

Adoptando estas medidas se obtendrían unos beneficios de **7.910€** mensuales. Teniendo en cuenta que los costes de mantenimiento del servidor son de 4.785,9€ aproximadamente, como se ha visto en el apartado *Fase 2: Mantenimiento del proyecto*, se obtendría un beneficio neto de 3.124,1€ al mes. Contando con que para albergar a



1.000 jugadores habría que ampliar la capacidad del servidor, se añade un gasto extra de 100€, por lo que el beneficio se reduce a aproximadamente **3.024,1€** mensuales. Según estos cálculos la inversión inicial del servidor descrita en el apartado *Fase 1: Creación del proyecto* (65.018,17€) se amortizará en 22 meses, momento en el que se empezarán a obtener ganancias.

## 12 Anexo A: Traducciones de las clases del Ragnarok <sup>[↑]</sup>

Generalmente todos los jugadores utilizan los nombres de las clases en inglés, ya que al no existir ningún servidor de Ragnarok de Gravity cuyo idioma sea el español, tampoco existe ninguna traducción oficial. Pese a ello se ha creado la *Tabla 22*: *Traducción inglés-español de las clases del Ragnarok* con posibles traducciones para cada job.

Inglés	Español
Novice	Novato
Swordman	Espadachín
Archer	Arquero
Acolyte	Acólito
Taekwon	Taekwon
Knight	Caballero
Lord knight	Señor caballero
Rune Knight	Caballero rúnico
Assassin	Asesino
Assassin cross	Cruz asesina
Guillotine cross	Guillotina asesina
Priest	Sacerdote
High priest	Sumo sacerdote
Arch bishop	Arzobispo
Wizard	Hechicero
High wizard	Archimago
Warlock	Conjurador
Blacksmith	Herrero
Whitesmith	Maestro herrero
Mechanic	Mecánico
Hunter	Cazador
Sniper	Tirador

Inglés	Español
Super novice	Super novato
Magician	Mago
Merchant	Mercader
Thief	Ladrón
Gunslinger	Pistolero
Crusader	Cruzado
Paladin	Paladín
Royal guard	Guardia Real
Rogue	Bribón
Stalker	Acechador
Shadow chaser	Cazador de sombras
Monk	Monje
Champion	Campeón
Shura	Shura
Sage	Sabio
Professor	Profesor
Sorcerer	Brujo
Alchemist	Alquimista
Creator	Creador
Genetic	Genético
Bard	Bardo
Clown	Payaso

Ranger	Guardabosques	Minstrel	Juglar
Dancer	Bailarina	Soul linker	Enlazador de almas
Gipsy	Cíngara	Star Gladiator	Gladiador de las estrellas
Wanderer	Nómada		

Tabla 22: Traducción inglés-español de las clases del Ragnarok

## 13 Anexo B: Estructura del foro [\[↑\]](#)

El foro está dividido en seis zonas, cada una con varios subforos:

- **Zona Staff**
  - General (Staff)
  - Donaciones (Staff)
  - Castigos y Denuncias (Staff)
  - Proyectos (Staff)
  - Bugs (Staff)
  - Guías y Apuntes (Staff)
  - Eventos (Staff)
  - Basurero (Staff)
  - Spam (Staff)
  - Bugtracker (Staff)
  - GMail

Esta zona es privada para los administradores, y los jugadores normales no pueden verla. A los nombres de los subforos se les ha añadido la palabra *Staff* para diferenciarlos de los públicos y así evitar confusiones. Cabe destacar el subforo *Bugtracker (Staff)*, donde está implementado el Bugtracker privado, y el subforo *Gmail*, donde llegan automáticamente los correos enviados por los usuarios mediante el formulario de contacto.

- **Tablón de Anuncios**
  - Noticias
  - Actualizaciones
  - Anuncios InGame
  - Información
  - Sanciones

Esta zona sólo tiene permisos de lectura para los jugadores, de forma que se utiliza por los administradores para notificar la información importante. Las publicaciones más recientes de *Noticias*, *Actualizaciones* y *Anuncions InGame*

aparecerán en la página principal, y las de este último subforo también se mostrarán dentro del juego la primera vez que un jugador entre con su personaje.

- **Presentaciones**

- Presentaciones y Despedidas

En esta zona los jugadores nuevos pueden presentarse, así como despedirse si los viejos van a dejar el servidor por un tiempo o de forma indefinida.

- **Soporte y Bugs**

- Soporte Técnico
- Ayuda InGame
- Dudas y Preguntas
- Sugerencias
- Bugtracker

En esta zona los jugadores exponen sus dudas y problemas de forma que un administrador u otro jugador pueda prestar su ayuda. También pueden publicar sugerencias para mejorar el servidor. Hay que destacar el subforo *Bugtracker*, donde se implementa el Bugtracker público.

- **Ragnarok Online**

- General
- War Of Emperium
- Mercado
- Guías y Builds
- Eventos
- Partys

Aquí se hablará de temas relacionados con el Ragnarok. El subforo *War of Emperium* es en cierto sentido especial porque generalmente sólo se abre un nuevo tópico al finalizar cada War of Emperium, donde los jugadores comentan las batallas.

- **Posada**

- Fotos, Screens y FanArt
- Vídeos del Servidor
- Fuera de Tema

En esta zona se puede hablar de cualquier cosa que no tenga que ver con el Ragnarok o de cosas que no tengan una implicación directa con el día a día dentro del juego.

## 14 Anexo C: Script de actualización de eAthena [\[↑\]](#)

A continuación se muestra el script que permite aplicar las actualizaciones del emulador eAthena oficial con el emulador modificado utilizado en BidaiRO:

```
#!/usr/bin/bash

BID_DIR=$1
EA_DIR=$2
REV_EA=$3
BID_SVN="svn checkout file:///home/svn/BidaiRO/trunk/"
#BID_SVN="svn checkout svn+ssh://kormath@localhost/BidaiRO/trunk/src"
EA_SVN="svn -r $REV_EA checkout http://eathena-
project.googlecode.com/svn/trunk/"
TMP="/tmp/merge$RANDOM"
PATCH=RagexeRE.patch
WD=`pwd`

if [ $# -ne 3 ]
then
    echo Uso:
    echo -e \\t sh merge.sh destino_bid destino_ea revision_ea
    echo -e \\t La revisión de eA es la última que se haya actualizado en
BidaiRO.

    exit
fi

if [ ! "${BID_DIR%/*}" ] || [ ! "${EA_DIR%/*}" ]
then
    echo "Los directorios de BidaiRO o eA no pueden empezar por / (deben
ser rutas relativas)"
    exit
fi

if [ -d $BID_DIR/.. ]
then
    rm -rf $BID_DIR
fi

if [ -d $EA_DIR/.. ]
then
    rm -rf $EA_DIR
fi

echo Obteniendo la revisión ${REV_EA} de eA:
$EA_SVN $EA_DIR

if [ $? -ne 0 ]
then
    echo "Error al obtener la revisión ${REV_EA} de eA."
    exit
fi

echo Obteniendo la última revisión de BidaiRO:
$BID_SVN $BID_DIR

if [ $? -ne 0 ]
```

```

then
    echo "Error al obtener la última revisión de BidaiRO."
    exit
fi
echo Copiando información de SVN de BidaiRO:
# Hacemos una copia de la información del SVN
mkdir $TMP
cp -R ${BID_DIR} ${TMP}

BID_SVN_DATA=`find ${TMP} -type d ! -regex .*svn/?.*`
for DATA in ${BID_SVN_DATA}
do
    rm -f ${DATA}/* 2> /dev/null

    sleep 0.2
done

echo Borrando el contenido de eA:
# Borramos todo el contenido de eA, salvo la información del SVN
EA_NOSVN_DATA=`find ${EA_DIR} -type d ! -regex .*svn/?.*`
for DATA in ${EA_NOSVN_DATA}
do
    rm -f ${DATA}/* 2> /dev/null

    sleep 0.2
done

echo Sustituyendo BidaiRO por eA:
# Copiamos los archivos de BidaiRO a eA
for DATA in ${EA_NOSVN_DATA}
do
    cp ${BID_DIR}%/*/${DATA}#*//* ${DATA} 2> /dev/null

    sleep 0.3
done

echo Actualizando a la última versión de eA
svn --accept postpone --force update $EA_DIR | tee log

echo Cambiando información del SVN de eA por el de BidaiRO
EA_SVN_DATA=`find ${EA_DIR} -type d -name .svn`
for DATA in ${EA_SVN_DATA}
do
    rm -rf ${DATA} 2> /dev/null

    cp -R ${TMP}/${BID_DIR}%/*/${DATA}#*/ ${DATA} 2> /dev/null

    sleep 0.5
done

echo Merge completo.
echo Recuerda resolver los conflictos y ejecutar los add/del necesarios.
echo El commit debe hacerse en $EA_DIR

```



## 15 Anexo D: Implementación del Trivial Online [\[↑\]](#)

Es posible jugar al trivial desde el servidor, mostrando las preguntas en la ventana de texto del juego y pudiendo contestarlas escribiendo la respuesta precedida de un carácter especial. El trivial está programado en C específicamente diseñado para funcionar en un servidor eAthena, por lo que aunque el código se ha diseñado para hacerlo lo más independiente posible, utiliza algunas de las funciones propias del emulador.

La forma de jugar al trivial es muy sencilla. Cada martes a las 22:00 se inicia una partida automática consistente en 150 preguntas aleatorias que se van sucediendo una por una. Cada pregunta tiene un tiempo límite durante el cual se puede responder, y se pasa a la siguiente cuando alguien acierta la respuesta o expira el tiempo. Poco antes de que finalice este tiempo se muestra una pista que consiste en la primera letra de cada palabra que componga la respuesta, así como el total de letras que componen la respuesta.

Para responder sólo hay que anteponer el carácter > a la respuesta, de forma que el servidor reconozca que el texto corresponde a una respuesta para el trivial. Una vez que alguien acierta, hay un segundo de pausa antes de que se cierre la posibilidad de responder a esa pregunta, por lo que es posible que más de un jugador contesten casi al mismo tiempo. El primero ganará 5 puntos, el segundo 3 y los restantes 1, y si se responde incorrectamente se pierde 1 punto. Al finalizar el trivial se mostrarán las cinco mejores puntuaciones, y los ganadores obtendrán dinero y experiencia, además de salir en el ranking del trivial disponible desde el Panel de Control.

En la *Imagen 39: Partida de Trivial Online* se puede observar una partida de ejemplo.

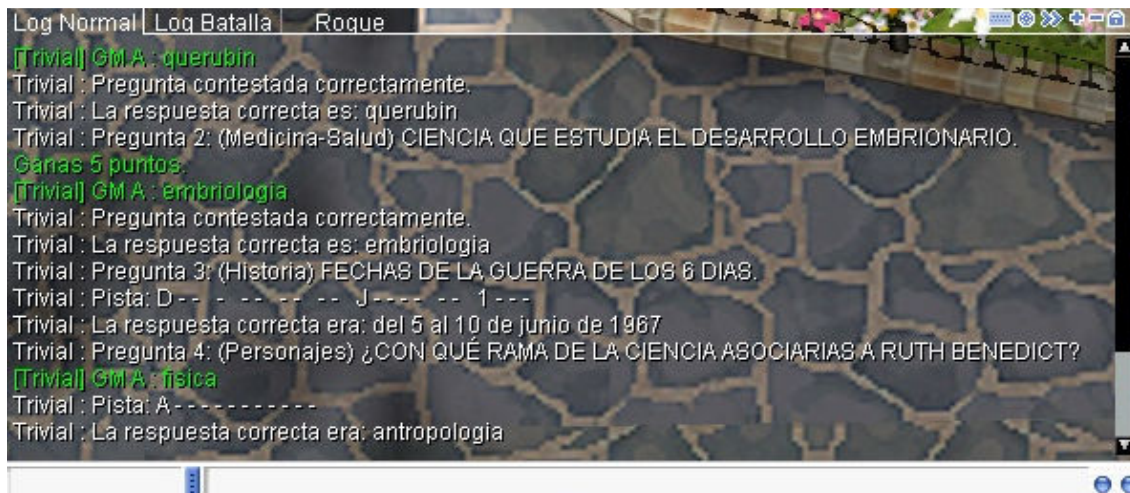


Imagen 39: Partida de Trivial Online

Esta forma de jugar al trivial es individual, pero también se ofrece un modo cooperativo entre clanes, en el cual sus miembros participan de forma conjunta y responden en nombre de su clan. También existe un comando **@trivial** que escribiéndolo permite activar o desactivar el trivial, de forma que no moleste a quien no quiera jugarlo.

El trivial es altamente configurable por parte de los administradores, pudiendo seleccionar el número de preguntas, las temáticas, los premios que obtendrá cada jugador dependiendo de la posición en la que acabe o incluso premios por cada respuesta contestada correctamente, entre otras cosas. Se disponen de 135.708 preguntas divididas en las siguientes categorías: arte, biología, canciones, ciencia, cine, cotilleo-TV, deporte, economía, eros-sex, física-química, gastronomía, geografía, historia, idiomas, informática, literatura, medicina-salud, miscelánea, mitología, música, personajes, política, refranes, refranes antiguos y Ragnarok.

El código del trivial está dividido en cuatro ficheros, como muestra la *Tabla 23: División en ficheros del código del trivial*.

Fichero	Descripción
<b>trivial.c</b>	Funciones principales, implementación de la lógica del trivial.
<b>trivial_handler.c</b>	Funciones para procesar las respuestas desde la ventana de texto y asignar las puntuaciones
<b>trivial_init.c</b>	Funciones de inicialización utilizadas al comenzar un nuevo trivial
<b>trivial.h</b>	Estructuras, constantes y definición de funciones y variables externas.

Tabla 23: División en ficheros del código del trivial

A continuación se describen brevemente las funciones utilizadas. El código completo puede consultarse en el *Anexo Digital B*.

- **trivial\_start**

Se encarga de comenzar un nuevo trivial. Llama a las funciones de inicialización para obtener las preguntas y aplicar la configuración inicial y avisa mediante un mensaje global en el juego de que ha comenzado el trivial.

- **trivial\_stop**

Es posible para un administrador finalizar un trivial manualmente, y entonces se llama a esta función. Básicamente lo que hace es comprobar que se esté jugando un trivial, y en caso afirmativo avisar de forma global de que un administrador lo ha abortado y llamar a la función *trivial\_free* para liberar los recursos.

- **send\_pregunta**

Mediante esta función se obtiene la siguiente pregunta de la lista y se imprime en la ventana de texto de cada jugador. La lista se habrá obtenido previamente al inicializar el trivial y se almacena en una tabla temporal<sup>[127]</sup> de la base de datos que viene a ser un subconjunto de la tabla principal donde se almacenan todas las preguntas disponibles.

- **send\_pista**

Al igual que la función *send\_pregunta*, esta se encarga de imprimir en la ventana de texto de cada jugador la pista correspondiente a la pregunta actual. Si la pista es un texto, mostrará la inicial de cada palabra y el número de letras que la componen, mientras que si es un número mostrará el rango de valores entre los que se puede encontrar la respuesta.

- **trivial\_core**

Esta función se ejecuta dentro de un bucle<sup>[128]</sup> por toda la duración del trivial, y se encarga de comprobar segundo a segundo el estado de la pregunta actual: si se ha contestado correctamente, si hay que mostrar una pista o si ha expirado el tiempo y hay que pasar a la siguiente pregunta. La implementación del bucle se lleva a cabo utilizando una función de eAthena de propósito general.

- **trivial\_ganadores**

Esta función se ejecuta una vez ha terminado la partida, y se encarga de mostrar a los ganadores en sus posiciones correspondientes. Además de eso, actualiza el ranking del Panel de Control y se encarga de llamar a la función de entrega de premios.

- **trivial\_premio**

Esta función simplemente se ocupa de llamar a otra función de entrega de premios más específica, dependiendo de si el trivial es de tipo individual o por clanes.

- **trivial\_premio\_individual**

Función para entregar los premios a un jugador específico para un trivial individual. Dependiendo de la posición en la que haya terminado y la configuración del trivial se le entregará un premio u otro.

- **trivial\_premio\_clan**

Función para entregar los premios a los miembros de un clan específico para un trivial por clanes. Dependiendo de la posición en la que haya terminado el clan y la configuración del trivial se les entregará un premio u otro.

- **trivial\_load\_puntuacion**

Es posible que un jugador se desconecte a mitad de una partida. Esta función cargará la puntuación que tenía antes de haber desconectado, de forma que pueda seguir jugando sin perder los puntos acumulados. La función sólo cargará la puntuación correspondiente al trivial que el jugador estaba jugando, si desconecta y más adelante entra a mitad de un trivial distinto no tendrá ningún punto.

- **trivial\_exp\_ponderada**

Cuando un jugador obtiene experiencia como premio, dependiendo de su nivel podrá ganar menos de la establecida para ese trivial. A medida que el nivel del jugador es más bajo, la experiencia recibida también descenderá. Esto es así porque a menor nivel menos experiencia se requiere para subir al siguiente, por lo que sería relativamente sencillo subir a personajes de nivel bajo a base de jugar trivials.

- **trivial\_free**

Esta función se encarga de liberar los recursos reservados para el trivial y ajustar el valor de ciertas variables de control.

- **trivial\_respuesta**

Mediante esta función el trivial procesa las respuestas de los jugadores y asigna sus puntuaciones dependiendo de si es correcta o no y de quién más haya acertado.

- **trivial\_puntuacion**

Esta función muestra la puntuación de un jugador cuando escribe el comando **@trivial puntuación**. Si no se utiliza ese comando, la única forma de saber los puntos obtenidos es esperar a la finalización del trivial.

- **set\_puntuacion**

Asigna una puntuación a un jugador o clan, dependiendo del tipo de trivial. Esta función se utiliza desde *trivial\_respuesta*.

- **get\_puntuacion**

Obtiene la puntuación de un jugador o clan. Devuelve un valor numérico para ser utilizado en otras partes del código y no debe confundirse con la función *trivial\_puntuacion*, que lo que hace es mostrar el valor por pantalla.

- **trivial\_core\_init\_preguntas**

Obtiene una lista aleatoria de preguntas para jugar en un trivial y las almacena en una tabla temporal. Dependiendo de la configuración, obtendrá las preguntas de toda la tabla o las filtrará por temática.

- **trivial\_init\_basico**

Debido a que es posible configurar muchas opciones, es posible iniciar un trivial con opciones por defecto (o configurar sólo algunas y dejar el resto también por defecto). Esta función se encarga de inicializar estas opciones, y se llamará siempre que se prepare un nuevo trivial.

El código del trivial contiene varias funciones parecidas a esta, cada una específica para configurar un elemento concreto. Dado que son todas muy similares entre sí, se omitirán en el presente documento.

## 16 Anexo E: Implementación de eventos [\[↑\]](#)

Es costumbre en BidaiRO hacer un evento especial para Halloween. El evento comienza unos días antes de esta fecha y termina el 31 de octubre, normalmente con una invasión a alguna ciudad consistente en varios monstruos disponibles en el juego y alguno personalizado. Los eventos de este tipo se programan en el lenguaje de scripting de eAthena. En este anexo se comentará en qué consistió el evento de Halloween de 2009, y todo el código está disponible en el *Anexo Digital C*.

Los eventos de varios días de duración consisten en la realización de varias pruebas que cada jugador debe resolver de forma individual. También pueden realizarse pruebas de forma colectiva, entre todo el servidor, pero no se suele optar por esto, aunque en 2008 se hizo así. Para este evento, el jugador debe encontrar varias piedras elementales repartidas a lo largo del mundo, y para ello un anciano monje le proporciona pistas sobre la localización de las mismas.

Cada piedra elemental le dará al jugador un objeto que deberá entregar al monje. Cuando los tenga todos, el monje podrá realizar un ritual para intentar eliminar el mal que se avecina. Este ritual se realizará siempre, independientemente de lo que hagan los jugadores, en un día y una hora establecida que el propio monje anunciará a los jugadores que le entreguen los objetos. Estos jugadores ganarán experiencia como premio por resolver el evento.

El ritual no saldrá del todo bien y aparecerán varios monstruos, uno de ellos más poderoso. Una vez derrotados, cuando parece que todo ha finalizado, aparecerán otros monstruos junto a uno de más poder, el verdadero causante de todo. Una vez derrotado, los jugadores que completaron el evento podrán hablar con el monje para recibir su experiencia.

Para llevar el flujo de este tipo de eventos se suelen utilizar dos tipos de variables, una global para todo el servidor y otra individual para cada jugador. El lenguaje de scripting de eAthena tiene muchos tipos de ámbitos<sup>[129]</sup> para las variables, por lo que es posible utilizar cada uno para unas tareas concretas. Es necesario saber por qué parte del evento va cada jugador para que el monje le mande buscar una piedra

elemental u otra, así como saber el estado global en el que se encuentra el evento (antes de comenzar la invasión, antes de que salga la primera tanda de monstruos, antes de que salga la segunda, todos los monstruos derrotados, etc.). Por ejemplo, la entrega de experiencia no debe hacerse hasta que no finalicen las invasiones (esto se controla con una variable global), y la experiencia sólo debe entregarse una vez a cada jugador y sólo a los que hayan completado el evento (esto se controla con una variable individual de jugador).

Además de este evento, también para Halloween de 2009 se implementó al vampiro Matías. El vampiro se encuentra en Prontera, la ciudad principal, y le pide al jugador 10 caramelos, algo típico de Halloween. A cambio y de forma aleatoria le entrega algún ítem o inicia una pequeña invasión. En la *Imagen 40: Evento del vampiro Matías* se puede ver al vampiro en la plaza de Prontera interactuando con un jugador.



Imagen 40: Evento del vampiro Matías



## 17 Anexo F: Pasos para la creación de un cliente [\[↑\]](#)

En la *Imagen 41: Diagrama para la creación de un cliente*, se muestran los pasos necesarios para crear un cliente compatible con el servidor BidaiRO, que se resumen a continuación:

- Obtención de un cliente genérico.
- Aplicación de parches para obtener un cliente personalizado.
- Inserción del anticheat.

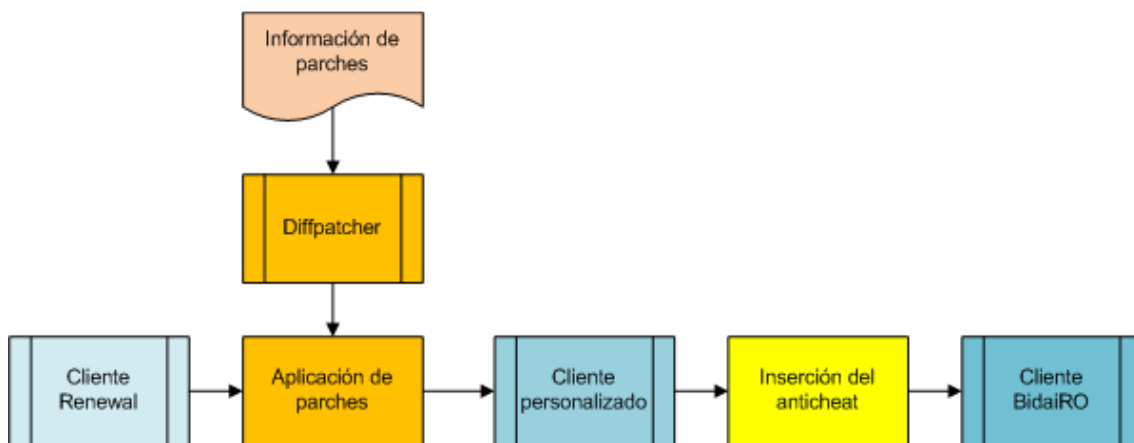


Imagen 41: Diagrama para la creación de un cliente

## 18 Anexo G: Implementación del parcheador [\[↑\]](#)

El parcheador está compuesto por dos bloques de código, uno principal donde se implementan casi todas las funcionalidades y la descarga e instalación de parches, y otro para la gestión de las personalizaciones visuales. Además de eso, utiliza una DLL necesaria para calcular los CRCs y tratar con los GRFs y los parches. Todo este código puede consultarse en el *Anexo Digital D*.

A continuación se describen los métodos<sup>[130]</sup> principales del primer bloque:

- **MainForm**

En este método se inicializan los componentes básicos de la aplicación, se comprueba si ya existe una instancia abierta (de forma que no se pueda abrir más de un parcheador al mismo tiempo), se crea el directorio donde se almacena la lista de actualizaciones instaladas (también se utiliza como temporal para descargarlas) y se asignan los manejadores<sup>[131]</sup>.

- **SetText**

Introduce un texto en las secciones informativas del parcheador, generalmente para indicar al usuario el estado de las actualizaciones y cualquier información útil.

- **SetEnabled**

Activa el botón *Jugar*, que por defecto está desactivado hasta que no se haya completado el proceso de actualización.

- **SetValue**

Mediante este método se actualiza la barra de progreso al descargar algún parche. Esta barra crece a medida que el parche se está descargando del servidor, y se completa cuando se ha recibido todo el fichero.

- **setVisible**

Muestra u oculta la barra de progreso, de forma que sólo sea visible cuando se está descargando algún parche.

- **GetUpdates**

Mediante este método se obtienen e instalan las actualizaciones. Primero se obtiene del servidor la lista de actualizaciones disponibles, y posteriormente se contrasta la versión de cada componente con la que tiene instalada el usuario, que está guardada en un fichero de texto dentro del directorio del parcheador creado en el método *MainForm* (ver el *Anexo H: Ficheros utilizados por el parcheador* para más información sobre el formato de las listas de versiones).

Si la versión instalada por el usuario es menor, se procede a la descarga del parche y a su instalación. Este método también se encarga de reiniciar el parcheador en caso de que lo requiera la actualización realizada. Durante todo este proceso se irá informando al usuario del estado de las actualizaciones y de si se han realizado correctamente o ha habido algún error, y al finalizar se activará el botón de *Jugar*.

- **updateGRF**

Llamado desde el método *GetUpdates*, se encarga de procesar la lista de versiones de los GRF y aplicar los parches en caso necesario. Se utiliza un método aparte para eso porque el formato que utilizan las actualizaciones de los GRF es ligeramente distinto al del resto de componentes.

- **checkPatchlist**

Comprueba si el formato de la lista de actualizaciones instaladas por el cliente es correcto, y en caso contrario lo repara. Este método se implementó para hacer frente a actualizaciones del parcheador que cambiaran el formato de esta lista, de forma que los usuarios que tuvieran

una versión antigua pudieran convertirla a la nueva sin necesidad de reinstalar todo el cliente.

- **Btn\_X\_Click**

Para cada botón existe un método asociado a un manejador que se encarga de realizar la acción correspondiente, ya sea iniciar el juego en el caso del botón *Jugar*, abrir alguna página web en el caso de los botones *Web*, *Foro* y *Regístrate*, o abrir la ventana de personalizaciones para el botón *Personaliza*. También existen otros métodos para cambiar el aspecto de los botones al pasar el ratón por encima y hacer clic en ellos.

- **WbrDocumentCompleted**

Se ejecuta cuando se ha terminado de cargar la página web principal del parcheador, y llama al método *GetUpdates* para comenzar con el proceso de actualización. Antiguamente este método también obtenía un código para realizar un seguimiento del usuario cuando el anticheat todavía no estaba implementado.

En la *Imagen 42: Mapa de métodos del parcheador (principal)* puede verse de forma visual la correspondencia de estos métodos con cada componente de la ventana principal.



Imagen 42: Mapa de métodos del parcheador (principal)

Los métodos del segundo bloque, correspondientes a las opciones de personalización, son los siguientes:

- **PersonalizaForm**

Inicializa los componentes básicos de la ventana de personalización, asigna los manejadores y carga la lista de personalizaciones de la web.

La carga se realiza de forma dinámica, de forma que para añadir o modificar contenido sólo hay que tocar la web, evitando tener que cambiar el código del parcheador. Consiste en una parte visual, donde se carga el contenido de la web correspondiente a la personalización que se quiere aplicar (por ejemplo, si se quiere cambiar el aura que obtiene un personaje al alcanzar el nivel 99, se carga una página donde pueden verse todas las auras disponibles), y en otra de información para el parcheador, donde lee un fichero con datos sobre cómo aplicar la personalización.

- **WbrDocumentCompleted**

Se ejecuta cuando se ha terminado de cargar la página correspondiente a la personalización que se quiere aplicar. Entonces se rellena un desplegable donde es posible elegir la personalización que se quiere aplicar, a partir del fichero de datos comentado anteriormente.

- **CmbEleccionSelectedIndexChanged**

Se ejecuta al seleccionar uno de los grupos de personalizaciones disponibles, y se encarga de cargar la web correspondiente a esas personalizaciones para previsualizarlas.

- **Btn\_X\_Click**

Para cada botón existe un método asociado a un manejador que se encarga de realizar la acción correspondiente, ya sea aplicar la personalización en el caso del botón *Aplicar*, o cerrar la ventana en el caso de *Cerrar*. También existen otros métodos para cambiar el aspecto de los botones al pasar el ratón por encima y hacer clic en ellos.

En la *Imagen 43: Mapa de métodos del parcheador (personalización)* puede verse de forma visual la correspondencia de estos métodos con cada componente de la ventana de personalización.

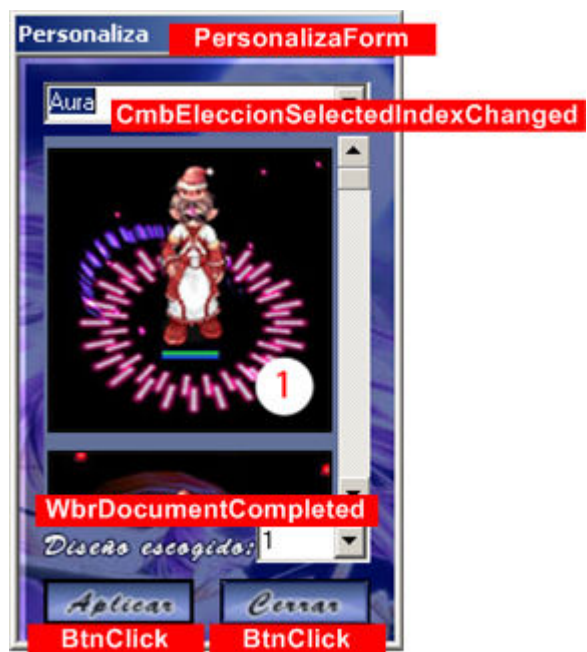


Imagen 43: Mapa de métodos del parcheador (personalización)

## 19 Anexo H: Ficheros utilizados por el parcheador

En el fichero **updates.txt**, obtenible a partir de la web del servidor, se encuentran las versiones de todos los componentes que se pueden actualizar. Las primeras cinco líneas indican la última versión de **BidaiPatch.exe**, **bgl.dll**, **BidaiRO.dll**, **anticheat.dll** y **BidaiRO.exe**, respectivamente. El resto de líneas indican el número de parche que se aplicará a **bidai2.grf** o **bidai.grf** (se componen de dos bloques delimitados por una **X**), la acción a realizar (A para añadir, B para borrar) y el nombre del fichero que contiene el parche. A continuación se muestra el contenido del fichero utilizado actualmente:

```
10
6
1
7
9
8 A idioma_1.gpf
9 A idioma_2.gpf
X
1 A bidai.grf
2 A navidad1.gpf
3 A navidad2.gpf
4 A navidad3.gpf
5 A navidad4.gpf
6 A navidad5.gpf
7 B antinavidad1.gpf
8 A antinavidad2.gpf
9 A 2009-01-20general.gpf
10 B 2009-01-20ctf.gpf
11 A 2009-01-24general.gpf
12 A 2009-02-15granja.gpf
13 A 2009-04-17loadings.gpf
14 A 2009-06-28cliente.gpf
15 A 2009-07-26ramas.gpf
16 A 2009-09-13RE_cartas.gpf
17 A 2009-07-11logo.gpf
18 B 2009-07-11RE_cartas_lista.gpf
19 A 2009-09-09headphones.gpf
20 A 2009-11-11manner.gpf
21 A navidad.gpf
22 A navidad.gpf
23 A navidad.gpf
24 A navidad.gpf
25 A navidad.gpf
26 B antinavidad1.gpf
27 A antinavidad2.gpf
28 A navidad.gpf
29 A 2009-12-24cliente.gpf
30 A 2009-12-25hotkeylub.gpf
31 A 2009-12-29Cliente.gpf
32 A 2009-12-31msgstringtable.gpf
```



33 B navidad.gpf  
34 B 2010-01-13antinavidad.gpf  
35 B navidad.gpf  
36 B 2010-01-13antinavidad.gpf  
37 A 2010-01-31gorros.gpf  
38 A 2010-02-04fullb.gpf  
39 A 2010-04-18gladiadores.gpf  
40 A 2010-05-01hats\_glad.gpf  
41 B 2010-05-08fogparam.gpf  
42 A 2010-05-08pvp99.gpf  
43 A 2010-06-01hats.gpf  
44 A 2010-06-05-hatsCorreccion.gpf  
45 A 2010-07-25-msgstringtable.gpf  
X

El fichero **patchbidai.txt** se almacena de forma local en el equipo de cada usuario y almacena la última versión descargada de cada componente, con la particularidad de que para los parches sólo se almacena una línea, ya que se descargarán todos los que encuentre a partir de esa versión. A continuación se muestra el contenido del fichero utilizado actualmente:

P10  
D6  
B1  
A7  
C9  
M9  
G45

## 20 Anexo I: Configuración del servidor MySQL [\[↑\]](#)

A continuación se explican las modificaciones realizadas en la configuración del servidor MySQL con respecto a los valores originales, y posteriormente se mostrará el fichero de configuración con estas modificaciones subrayadas para su fácil visualización.

- **max\_allowed\_packet = 64M**

Esta variable se encarga de controlar el tamaño máximo que puede tener un paquete de datos. Si el paquete es mayor que el tamaño aquí indicado, el servidor lanzará un error y cerrará la conexión, por lo que interesa aumentarlo para evitar esto. Además el consumo de memoria sólo se incrementa cuando es necesario, por lo que es eficiente.

- **max\_heap\_table\_size = 128M**

Esta variable controla el máximo de memoria que se le puede asignar a las tablas antes de empezar a usar el disco duro. Utilizar tablas en memoria es mucho más eficiente dada la rapidez de acceso a la información comparado con el disco duro, pero hay que ajustar este valor con cuidado para no dejar al sistema sin memoria.

- **tmp\_table\_size = 128M**

Esta variable funciona igual que la *max\_heap\_table\_size* descrita anteriormente, pero para tablas temporales. Es aconsejable utilizar el mismo valor en las dos variables.

- **table\_cache = 1024**

El número máximo de tablas que puede tener el servidor MySQL en todos los hilos<sup>[132]</sup>. Para ajustarlo hay que comprobar periódicamente el número de tablas que se tienen abiertas, y si se acerca al valor máximo, debe aumentarse.

- **wait\_timeout = 3600**

Aquí se especifican el número de segundos que esperará el servidor para cerrar una conexión no interactiva<sup>[133]</sup> si no ha habido actividad.

- **interactive\_timeout = 3600**

Aquí se especifican el número de segundos que esperará el servidor para cerrar una conexión interactiva<sup>[134]</sup> si no ha habido actividad.

- **join\_buffer\_size = 1M**

Cuanto más grande sea el valor de esta variable, más rápido se procesarán las joins<sup>[135]</sup> entre tablas que no utilicen índices. Lo ideal es utilizar los índices<sup>[136]</sup>, pero no siempre es posible, por lo que es necesario ajustar esta variable para esos casos.

- **query\_cache\_limit = 2M**

Esta variable asigna el tamaño máximo de bytes que puede tener el resultado de una consulta para que se guarde en caché<sup>[137]</sup>. Si supera el máximo no se guardará.

- **query\_cache\_size = 64M**

Esta variable indica el tamaño máximo de la caché para almacenar los resultados de las consultas. Hay consultas que se realizan muchas veces, y para estos casos es aconsejable poder almacenar los resultados para reutilizarlos y ganar en velocidad.

- **sort\_buffer\_size = 2M**

Aquí se indica el tamaño máximo del buffer<sup>[138]</sup> utilizado para tratar con operaciones de ordenación<sup>[139]</sup> y agrupación<sup>[140]</sup> (*order by* y *group by*). Este buffer ayuda a mejorar consultas de este tipo que no utilicen índices o que no estén optimizadas, pero asignarle un valor demasiado alto puede afectar al rendimiento del servidor MySQL, por lo que debe utilizarse con cuidado.

- **read\_buffer\_size = 2M**

Esta variable especifica el tamaño máximo del buffer que se utilizará cuando sea necesario leer una tabla de forma secuencial.

- **read\_rnd\_buffer\_size = 4M**

Esta variable especifica el tamaño máximo del buffer que se utilizará para leer filas que hayan sido ordenadas previamente mediante una sentencia *order by*.

- **key\_buffer\_size = 64M**

Esta variable asigna el tamaño máximo del buffer utilizado para los índices de las tablas. Es una de las variables más importantes, ya que impacta en la rapidez con la que el servidor devuelve los resultados de las consultas.

- **log\_slow\_queries = /home/mysql\_log/mysql-slow.log**

Esta variable especifica el fichero donde se almacenarán las consultas consideradas lentas. Es útil para analizar qué consultas están empleando más tiempo en realizarse y así poder optimizarlas.

- **long\_query\_time = 2**

Esta variable está relacionada con el *log\_slow\_queries* descrito anteriormente, y especifica el número mínimo de segundos que debe emplear una consulta para considerarse lenta y ser almacenada en el log.

## El fichero de configuración es el siguiente:

```
# The MySQL database server configuration file.
#
# You can copy this to one of:
# - "/etc/mysql/my.cnf" to set global options,
# - "~/.my.cnf" to set user-specific options.
#
# One can use all long options that the program supports.
# Run program with --help to get a list of available options and with
# --print-defaults to see which it would actually understand and use.
#
# For explanations see
# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

# This will be passed to all mysql clients
# It has been reported that passwords should be enclosed with ticks/quotes
# especially if they contain "#" chars...
# Remember to edit /etc/mysql/debian.cnf when changing the socket location.
[client]
port                = 3306
socket              = /var/run/mysqld/mysqld.sock

# Here is entries for some specific programs
# The following values assume you have at least 32M ram

# This was formally known as [safe_mysqld]. Both versions are currently
# parsed.
[mysqld_safe]
socket              = /var/run/mysqld/mysqld.sock
nice                = 0

[mysqld]
#
# * Basic Settings
#
#
# * IMPORTANT
#   If you make changes to these settings and your system uses apparmor, you
#   may
#   also need to also adjust /etc/apparmor.d/usr.sbin.mysqld.
#
user                = mysql
pid-file            = /var/run/mysqld/mysqld.pid
socket              = /var/run/mysqld/mysqld.sock
port                = 3306
basedir             = /usr
datadir             = /var/lib/mysql
tmpdir              = /tmp
language            = /usr/share/mysql/english
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address        = 127.0.0.1
#
# * Fine Tuning
#
```

```

key_buffer                = 16M
max_allowed_packet        = 64M
max_heap_table_size       = 128M
tmp_table_size            = 128M
thread_stack              = 128K
thread_cache_size         = 8
table_cache               = 1024
wait_timeout              = 3600
interactive_timeout        = 3600
join_buffer_size          = 1M
#
# * Query Cache Configuration
#
query_cache_limit          = 2M
query_cache_size           = 64M
sort_buffer_size           = 2M
read_buffer_size          = 2M
read_rnd_buffer_size       = 4M
key_buffer_size           = 64M
#
# * Logging and Replication
#
# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
#log                        = /var/log/mysql/mysql.log
#
# Error logging goes to syslog. This is a Debian improvement :)
#
#log_warnings = 2
# Here you can see queries with especially long duration
#log_slow_queries          = /var/log/mysql/mysql-slow.log
log_slow_queries           = /home/mysql_log/mysql-slow.log
long_query_time = 2
#log-queries-not-using-indexes
#
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
#       other settings you may need to change.
#server-id                 = 1
#log_bin                   = /var/log/mysql/mysql-bin.log
#
#log_bin                   = /home/mysql_log/mysql-bin.log
#expire_logs_days          = 10
#max_binlog_size           = 100M
#binlog_do_db              = ragnarok
#
#binlog_ignore_db          = include_database_name
#
# * BerkeleyDB
#
# Using BerkeleyDB is now discouraged as its support will cease in 5.1.12.
skip-bdb
#
# * InnoDB
#
# InnoDB is enabled by default with a 10MB datafile in /var/lib/mysql/.
# Read the manual for more InnoDB related options. There are many!
# You might want to disable InnoDB to shrink the mysqld process by circa
100MB.
skip-innodb
#

```

```

# * Security Features
#
# Read the manual, too, if you want chroot!
# chroot = /var/lib/mysql/
#
# For generating SSL certificates I recommend the OpenSSL GUI "tinyca".
#
# ssl-ca=/etc/mysql/cacert.pem
# ssl-cert=/etc/mysql/server-cert.pem
# ssl-key=/etc/mysql/server-key.pem

[mysqldump]
quick
quote-names
max_allowed_packet      = 16M

[mysql]
#no-auto-rehash # faster start of mysql but no tab completion

[isamchk]
key_buffer              = 16M

#
# * NDB Cluster
#
# See /usr/share/doc/mysql-server-*/README.Debian for more information.
#
# The following configuration is read by the NDB Data Nodes (ndbd processes)
# not from the NDB Management Nodes (ndb_mgmd processes).
#
# [MYSQL_CLUSTER]
# ndb-connectstring=127.0.0.1

#
# * IMPORTANT: Additional settings that can override those from this file!
# The files must end with '.cnf', otherwise they'll be ignored.
#
!includedir /etc/mysql/conf.d/

```

## 21 Anexo J: Análisis de rendimiento de MySQL [\[↑\]](#)

La ejecución de **tuning-primer.sh** muestra lo siguiente:

```
-- MYSQL PERFORMANCE TUNING PRIMER --
- By: Matthew Montgomery -

MySQL Version 5.0.67-0ubuntu6-log x86_64

Uptime = 505 days 23 hrs 39 min 47 sec
Avg. qps = 13
Total Questions = 592113593
Threads Connected = 21

Server has been running for over 48hrs.
It should be safe to follow these recommendations

To find out more information on how each of these
runtime variables effects performance visit:
http://dev.mysql.com/doc/refman/5.0/en/server-system-variables.html
Visit http://www.mysql.com/products/enterprise/advisors.html
for info about MySQL's Enterprise Monitoring and Advisory Service

SLOW QUERIES
The slow query log is enabled.
Current long_query_time = 2 sec.
You have 4187 out of 592113614 that take longer than 2 sec. to complete
Your long_query_time seems to be fine

BINARY UPDATE LOG
The binary update log is NOT enabled.
You will not be able to do point in time recovery
See http://dev.mysql.com/doc/refman/5.0/en/point-in-time-recovery.html

WORKER THREADS
Current thread_cache_size = 8
Current threads_cached = 1
Current threads_per_sec = 0
Historic threads_per_sec = 0
Your thread_cache_size is fine

MAX CONNECTIONS
Current max_connections = 100
Current threads_connected = 21
Historic max_used_connections = 25
The number of used connections is 25% of the configured maximum.
Your max_connections variable seems to be fine.

MEMORY USAGE
Max Memory Ever Allocated : 366 M
Configured Max Per-thread Buffers : 912 M
Configured Max Global Buffers : 138 M
Configured Max Memory Limit : 1 G
Physical Memory : 1.93 G
Max memory limit seem to be within acceptable norms
```



#### KEY BUFFER

Current MyISAM index space = 432 M  
Current key\_buffer\_size = 64 M  
Key cache miss rate is 1 : 405  
Key buffer fill ratio = 100.00 %  
You could increase key\_buffer\_size  
It is safe to raise this up to 1/4 of total system memory;  
assuming this is a dedicated database server.

#### QUERY CACHE

Query cache is enabled  
Current query\_cache\_size = 64 M  
Current query\_cache\_used = 21 M  
Current query\_cache\_limit = 2 M  
Current Query cache Memory fill ratio = 33.53 %  
Current query\_cache\_min\_res\_unit = 4 K  
MySQL won't cache query results that are larger than query\_cache\_limit in size

#### SORT OPERATIONS

Current sort\_buffer\_size = 2 M  
Current read\_rnd\_buffer\_size = 4 M  
Sort buffer seems to be fine

#### JOINS

Current join\_buffer\_size = 1.00 M  
You have had 41589 queries where a join could not use an index properly  
You should enable "log-queries-not-using-indexes"  
Then look for non indexed joins in the slow query log.  
If you are unable to optimize your queries you may want to increase your  
join\_buffer\_size to accommodate larger joins in one pass.

Note! This script will still suggest raising the join\_buffer\_size when  
ANY joins not using indexes are found.

#### OPEN FILES LIMIT

Current open\_files\_limit = 2158 files  
The open\_files\_limit should typically be set to at least 2x-3x  
that of table\_cache if you have heavy MyISAM usage.  
Your open\_files\_limit value seems to be fine

#### TABLE CACHE

Current table\_cache value = 1024 tables  
You have a total of 348 tables  
You have 550 open tables.  
The table\_cache value seems to be fine

#### TEMP TABLES

Current max\_heap\_table\_size = 128 M  
Current tmp\_table\_size = 128 M  
Of 1512771 temp tables, 30% were created on disk  
Perhaps you should increase your tmp\_table\_size and/or max\_heap\_table\_size  
to reduce the number of disk-based temporary tables  
Note! BLOB and TEXT columns are not allow in memory tables.  
If you are using these columns raising these values might not impact your  
ratio of on disk temp tables.

#### TABLE SCANS

Current read\_buffer\_size = 2 M  
Current table scan ratio = 466 : 1  
read\_buffer\_size seems to be fine

#### TABLE LOCKING

Current Lock Wait ratio = 1 : 42070

Your table locking seems to be fine

Todo está correcto, por lo que se procede a ejecutar **mysqltuner.pl**:

```
----- General Statistics -----
[--] Skipped version check for MySQLTuner script
[OK] Currently running supported MySQL version 5.0.67-0ubuntu6-log
[OK] Operating on 64-bit architecture

----- Storage Engine Statistics -----
[--] Status: +Archive -BDB -Federated -InnoDB -ISAM -NDBCluster
[--] Data in MyISAM tables: 1G (Tables: 331)
[!!] Total fragmented tables: 34

----- Performance Metrics -----
[--] Up for: 505d 23h 43m 27s (592M q [13.544 qps], 96M conn, TX: 217B, RX: 56B)
[--] Reads / Writes: 6% / 94%
[--] Total buffers: 266.0M global + 9.1M per thread (100 max threads)
[OK] Maximum possible memory usage: 1.2G (59% of installed RAM)
[OK] Slow queries: 0% (4K/592M)
[OK] Highest usage of available connections: 25% (25/100)
[OK] Key buffer size / total MyISAM indexes: 64.0M/432.4M
[OK] Key buffer hit rate: 99.8% (6B cached / 15M reads)
[OK] Query cache efficiency: 39.6% (19M cached / 48M selects)
[OK] Query cache prunes per day: 29
[OK] Sorts requiring temporary tables: 0% (132 temp sorts / 4M sorts)
[OK] Temporary tables created on disk: 23% (457K on disk / 1M total)
[OK] Thread cache hit rate: 99% (2K created / 96M connections)
[!!] Table cache hit rate: 3% (550 open / 14K opened)
[OK] Open file limit used: 41% (901/2K)
[OK] Table locks acquired immediately: 99% (669M immediate / 669M locks)

----- Recommendations -----
General recommendations:
  Run OPTIMIZE TABLE to defragment tables for better performance
  Increase table_cache gradually to avoid file descriptor limits
Variables to adjust:
  table_cache (> 1024)
```

Sugiere aumentar la caché de las tablas y desfragmentarlas. Sobre lo primero, el script anterior indica que los valores son correctos y no se ha observado ningún problema en el uso diario, por lo que se dejará como está. Para obtener las tablas fragmentadas, se ejecuta un script que calcula la fragmentación y se planifica su optimización para el siguiente reinicio por mantenimiento del servidor. La salida del script es la siguiente:

```
dev_ragnarok.mapreg is 4% fragmented.
mysql.proc is 10% fragmented.
mysql.procs_priv is 20% fragmented.
mysql.user is 8% fragmented.
ragnarok.anticheat is 3% fragmented.
ragnarok.cart_inventory is 6% fragmented.
ragnarok.guild is 1% fragmented.
```

ragnarok.guild\_alliance is 9% fragmented.  
ragnarok.guild\_member is 2% fragmented.  
ragnarok.ipbanlist is 100% fragmented.  
ragnarok.mail is 3% fragmented.  
ragnarok.mapreg is 2% fragmented.  
ragnarok.mercenary is 33% fragmented.  
ragnarok.party is 1% fragmented.  
ragnarok.sc\_data is 2% fragmented.  
ragnarok\_foro.phpbb\_banlist is 28% fragmented.  
ragnarok\_foro.phpbb\_confirm is 65% fragmented.  
ragnarok\_foro.phpbb\_search\_results is 95% fragmented.  
ragnarok\_foro.phpbb\_sessions is 78% fragmented.  
ragnarok\_log.anticheatlog is 69% fragmented.  
ragnarok\_log.contestador is 100% fragmented.  
ts.ts2\_bans is 100% fragmented.  
ts3.channel\_properties is 2% fragmented.  
ts3.clients is 11% fragmented.  
ts3.perm\_server\_group is 2% fragmented.  
ts3.tokens is 100% fragmented.

## 22 Diccionario de términos y acrónimos

- [1] **MMORPG:** Acrónimo de Massively Multiplayer Online Role-Playing Game (juego de rol multijugador online masivo), un género de videojuegos en el que un gran número de jugadores interactúan entre ellos en un mundo virtual.
- [2] **Rate:** En español, sinónimo de *velocidad*. Define el multiplicador aplicado a la ganancia de experiencia y a la probabilidad de obtener objetos en un servidor de Ragnarok Online con respecto al servidor oficial, que se considera **x1**. Se dividen en tres tipos, **low rates**, **medium rates** y **high rates** indicando un tipo de velocidad lenta, media y rápida, respectivamente. Normalmente se consideran low rates de **x1** a **x10**, medium rates de **x11** a **x50**, y high rates de **x51** en adelante.
- [3] **Host:** Un servicio de hosting es aquel que corre servidores de Internet para permitir a organizaciones o individuos compartir y ofrecer sus propios contenidos en la Red.
- [4] **Emulador:** Software que duplica las funciones de un sistema en otro sistema diferente, de forma que se comporta como el primero.
- [5] **SVN:** Acrónimo de Subversion<sup>[7]</sup>, sistema de control de versiones<sup>[8]</sup> de Apache.
- [6] **Entorno de desarrollo:** Entorno utilizado por los desarrolladores para implementar nuevas funcionalidades, hacer pruebas y depurar.
- [7] **Entorno de producción:** Entorno utilizado por el usuario final. Todo lo que llega a este entorno ha debido pasar previamente por un entorno de desarrollo y haber sido probado para minimizar el riesgo de errores.
- [8] **Evento:** Algo especial que sucede en el juego de forma esporádica o cada cierto tiempo determinado, como una invasión de monstruos a una ciudad, un trivial, etc.
- [9] **MIT:** Massachusetts Institute of Technology (Instituto de Tecnología de Massachusetts).
- [10] **Arcade:** En España se la conoce también como *máquina recreativa* o simplemente *recreativa*, se trata de un videojuego situado en algún lugar público al que se puede jugar introduciendo una moneda. El interfaz de juego

- consistía generalmente en un joystick y uno o más botones, pero también podía ser un volante o algo similar dependiendo del tipo de juego.
- [11] **Jumper:** Sobre todo utilizado en informática, se trata de un pequeño conductor para cerrar un circuito eléctrico.
  - [12] **PC compatible:** A principios de la década de los 80, IBM creó una serie de ordenadores personales de bajo coste. Los PCs compatibles eran clones de las máquinas de IBM, y a partir de ellos surgieron la mayoría de los ordenadores que se utilizan hoy en día para tareas generales.
  - [13] **Pad:** También llamado simplemente *mando*, es un dispositivo de entrada utilizado en las consolas diseñado para jugar con ambas manos. Originalmente consistía en una cruz o circunferencia de 4 u 8 posiciones en el lado izquierdo para realizar movimientos y dos o más botones en el derecho para realizar acciones, pero han ido evolucionando y ahora pueden tener pequeños joysticks, más botones, vibración, etc.
  - [14] **LCD:** Acrónimo de Liquid Cristal Display (*pantalla de cristal líquido*), se trata de una pantalla fina electrónica que utiliza las propiedades de modulación de luz de los cristales líquidos (cristales con características entre las de un líquido convencional y un cristal sólido) para mostrar imágenes.
  - [15] **2D:** Dos dimensiones.
  - [16] **3D:** Tres dimensiones.
  - [17] **Shareware:** Software propietario ofrecido al usuario de forma gratuita con características limitadas, que son desbloqueadas una vez se paga el precio del producto.
  - [18] **LAN:** Acrónimo de Local Area Network (Red de Área Local), red de cobertura pequeña como una casa, una oficina o un grupo pequeño de edificios.
  - [19] **CD:** Acrónimo de Compact Disc (*disco compacto*), es un dispositivo óptico para almacenar datos digitales.
  - [20] **MUD:** Acrónimo de Multi-User Dungeon (Mazmorra Multijugador), son los precursores de los MMOPRGs. Vienen a ser un juego online multijugador en tiempo real pero en modo texto, sin imágenes ni animaciones.
  - [21] **D&D:** Acrónimo de Dungeons & Dragons, juego de rol publicado en 1974 y muy jugado a día de hoy.

- [22] **ARPANET**: Acrónimo de Advanced Research Projects Agency Network (Red de la Agencia de Investigación de Proyectos Avanzados), la primera red operacional de intercambio de paquetes y precursora de Internet.
- [23] **PVP**: Acrónimo de Player Vs. Player (jugador contra jugador), se utiliza para definir un juego online en el que se combate contra otros jugadores en vez de o además de contra la máquina.
- [24] **PK**: Acrónimo de Player Killing (matar jugadores), es equivalente a PvP pero más utilizado en MUDs. En MMORPGs se puede considerar como un combate contra otros jugadores sin el consentimiento de todas las partes implicadas.
- [25] **RP**: Acrónimo de Roleplaying (jugar siguiendo un rol), indica que el jugador debe actuar siguiendo el rol de su personaje.
- [26] **Trigger**: Del inglés *disparador*, es un evento utilizado en bases de datos que se activa al realizarse una acción (inserción, actualización o borrado de una o varias filas) y ejecuta una serie de acciones definidas por el programador, generalmente relativas a los datos con los que se acaba de operar.
- [27] **Script**: Programas informáticos interpretados generalmente sencillos almacenados en texto plano.
- [28] **PLATO**: Acrónimo de Programmed Logic for Automated Teaching Operations (Lógica Programada para Operaciones de Enseñanza Automatizada), sistema en red creado en la Universidad de Illinois.
- [29] **X-WING**: Simulador de vuelo de 1993 basado en la trilogía de Star Wars.
- [30] **Estrategia por turnos**: Juego donde el jugador y sus oponentes (sean artificiales o humanos) pueden realizar un número limitado de acciones en su turno, momento en el cual lo finalizan y dan paso al turno del jugador siguiente, hasta que todos han jugado y se vuelve a empezar.
- [31] **Heroes of Might and Magic V**: Quinta entrega del popular juego de estrategia por turnos Heroes of Might and Magic, de New World Computing, en el que se controla a un héroe encargado de recolectar recursos y conquistar ciudades y castillos para poder crear ejércitos y obtener nuevas habilidades.
- [32] **Java**: Lenguaje de programación orientado a objetos, creado por Sun Microsystems en 1995.
- [33] **Flash**: Plataforma multimedia de Adobe para añadir animaciones, vídeos e interactividad a las páginas web.

- [34] **PJ:** Acrónimo de Personaje Jugador, viene a ser el personaje que controla el jugador en un mundo de fantasía.
- [35] **RPG:** Acrónimo de Rol Playing Game (Juego de Rol).
- [36] **Nivel:** Mide lo competente que es un PJ en los aspectos más importantes del juego. Se empieza en nivel 1 y aumentando de niveles es posible mejorar las características del PJ. Generalmente existe un tope a partir del cual no se puede seguir subiendo; en el caso del Ragnarok Online es el nivel 99, momento en el cual es posible recomenzar desde nivel 1 y volver a subir a 99 con características mejoradas. En una actualización reciente se ha aumentado el límite a 150.
- [37] **Puntos de experiencia:** Para cuantificar el esfuerzo necesario para subir un nivel se utilizan los puntos de experiencia. Cada nivel requiere un determinado número de puntos de experiencia que generalmente se obtienen matando enemigos o realizando otras acciones. Normalmente la experiencia necesaria para subir un nivel es creciente, por lo que cuesta más alcanzar los niveles superiores, como es el caso del Ragnarok Online.
- [38] **Modelo cliente-servidor:** Tipo de arquitectura distribuida en la que la carga de trabajo se divide entre un servidor que proporciona un servicio y un cliente que lo solicita. El servidor comparte los recursos con los clientes que se conectan a él para solicitárselos.
- [39] **Pack de expansión:** Contenido extra que se añade al cliente del MMORPG para obtener acceso a nuevas zonas, equipo, habilidades y cualquier cosa que se decida actualizar. En algunos juegos se compran y en otros son gratuitos.
- [40] **Clan:** Grupo de jugadores aliados que se ayudan mutuamente para conseguir ciertos objetivos comunes o individuales.
- [41] **Warp:** Comunicación entre mapas que permite a los jugadores moverse de uno a otro, generalmente adyacentes.
- [42] **Stat:** Atributos que el jugador puede asignar al personaje para hacerlo más competente a la hora de realizar ciertas acciones. Los atributos disponibles son fuerza, agilidad, vitalidad, inteligencia, destreza y suerte.
- [43] **Flee:** En español *esquiva*, es la facilidad que tiene un personaje de evitar ser golpeado por un golpe físico. Depende principalmente de la agilidad, de la destreza del atacante y del número de oponentes que estén atacando (cuantos más sean más difícil será esquivar).

- [44] **Puntos de vida o HP:** El HP, del inglés Hit Point, mide la capacidad del personaje de resistir ataques antes de morir.
- [45] **Ítems de curación:** Pociones, comidas o cualquier cosa que el personaje pueda tomar de forma voluntaria para recuperar una parte de la vida perdida.
- [46] **Regeneración de HP:** Rapidez con la que los puntos de vida aumentan con el paso del tiempo hasta alcanzar el máximo de ese jugador.
- [47] **Puntos de habilidad o SP:** El SP, del inglés Skill Point, indica cuántas habilidades o hechizos puede utilizar el personaje. No debe confundirse con los puntos de habilidad<sup>[55]</sup> que el jugador puede repartir entre las habilidades y los hechizos para mejorarlos
- [48] **Ítems de recuperación de SP:** Pociones, comidas o cualquier cosa que el personaje pueda tomar de forma voluntaria para recuperar una parte de los puntos de habilidad gastados.
- [49] **Regeneración de SP:** Rapidez con la que los puntos de habilidad aumentan con el paso del tiempo hasta alcanzar el máximo de ese jugador.
- [50] **Cast time:** En español podría traducirse como *tiempo de preparación*, aunque en el Ragnarok se utiliza la expresión *tiempo de casteo*. Es el tiempo que necesita una habilidad desde que se quiere utilizar hasta que hace efecto, y puede ser desde inmediato hasta durar varios segundos. La duración se puede reducir en casi todos los casos principalmente aumentando la destreza, pero también mediante habilidades u objetos. Hay algunas habilidades cuyo tiempo de preparación no se puede reducir.
- [51] **Golpe crítico:** También llamado simplemente *crítico*, es un golpe que acierta siempre y realiza el máximo daño posible, independientemente de la defensa del objetivo.
- [52] **Perfect dodge:** En español *esquiva perfecta*, es la probabilidad de esquivar completamente un golpe físico. No es el mismo tipo de esquiva que proporciona la agilidad.
- [53] **Clase o job:** Tipo de rol que desempeñará el personaje en el juego, que definirá sus habilidades y sus puntos fuertes y debilidades.
- [54] **Nivel de job:** Al igual que existe un nivel base que determina lo bueno que es el jugador en general, el job también tiene un sistema de niveles que indican la competencia del personaje en su clase. El nivel máximo es 9, 50, 70 ó 90, dependiendo del job.



- [55] **Puntos de habilidad:** Puntos necesarios para subir el nivel de una habilidad, obtenidos al subir un nivel de job. No debe confundirse con el SP<sup>[47]</sup>.
- [56] **Clase primaria o primer job:** Jobs que pueden escogerse cuando se alcanza el nivel 9 de novice.
- [57] **Clase secundaria o segundo job:** Jobs que pueden escogerse entre los niveles 40 y 50 de primer job.
- [58] **Renacer:** Una vez alcanzado el máximo nivel, el jugador puede volver a rehacer su personaje desde nivel 1 con varias bonificaciones.
- [59] **Clase extendida:** Job correspondiente al segundo job una vez que el personaje ha renacido.
- [60] **Clase terciaria o tercer job:** Jobs que pueden escogerse una vez alcanzado el nivel 70 de segundo job.
- [61] **Grifo:** Animal mitológico resultante de un cruce entre un águila y un león.
- [62] **Chat global:** Implementación personalizada de BidaiRO (no existe en los servidores oficiales), permite que un jugador se comuniquen con el resto del servidor esté donde esté. Es posible desactivar el canal de chat de forma individual si no se quiere ser molestado. Para evitar malos usos, es necesario tener un nivel mínimo para comenzar a utilizarlo, y hay que esperar 20 segundos para poder volver a usarlo.
- [63] **Título:** Rango que se le asigna a uno o más miembros de un clan. Pueden crearse hasta 20 títulos, por lo que lo más normal es que se les asigne a grupos de jugadores, ya que un clan puede llegar a tener hasta 76 miembros.
- [64] **Guild Master o GM:** Podría traducirse como Señor del Clan, es el jugador fundador del clan y tiene privilegios para llevar a cabo todas las acciones que pueden realizarse dentro de él, como son incluir y expulsar miembros y gestionar distintos aspectos.
- [65] **Tasa:** Porcentaje de la experiencia base (la de job no computa) que donará un jugador a su clan para que suba de nivel. Las tasas se asocian a los títulos de los clanes, por lo que hay que asignarlas pensando que en la mayoría de los casos afectará a más de un jugador. La tasa mínima es de 0 (no dona nada) y la máxima de 50 (dona la mitad de la experiencia ganada).
- [66] **Skill activa/pasiva:** Una skill activa es aquella que no aporta ningún beneficio hasta que se utiliza explícitamente, consumiendo una cantidad de SP, y generalmente tiene una duración determinada para luego dejar de hacer efecto.

Las skills pasivas siempre están activas, no hay que usarlas de forma explícita y no consumen SP.

- [67] **WoE**: Acrónimo de War of Emperium, en español *Guerra del Emperium*, es un evento que se celebra dos veces por semana durante el cual los clanes pueden luchar entre sí para conquistar uno o más castillos y así ganar prestigio y premios.
- [68] **Beta**: Fase del desarrollo de software que generalmente consiste en probar dicho software utilizándolo normalmente antes de entregarlo al usuario final, de forma que puedan detectarse y corregirse fallos y sea posible ajustar y mejoren las características que se vayan considerando necesarias. Las *betas cerradas* las prueba un número reducido y escogido de usuarios, mientras que las *betas abiertas* las prueba un número mucho más amplio, en muchos casos el propio usuario final.
- [69] **Manga**: Palabra japonesa para designar una historieta en general. Fuera de Japón se utiliza para referirse exclusivamente a los cómics japoneses.
- [70] **Banner**: Formato publicitario de Internet consistente en una imagen o animación en una página web, diseñadas para atraer la atención del navegante y que al hacer clic en él le lleve a la página anunciada.
- [71] **Proxy**: Programa o dispositivo informático que realiza acciones en representación de otro. Los hay con distintos grados de anonimidad, algunos son transparentes y es posible identificar quién lo está usando, pero otros son completamente anónimos.
- [72] **HTML**: Acrónimo de HyperText Markup Language (Lenguaje de Marcado de Hipertexto), es el principal lenguaje de marcas utilizado para realizar páginas web.
- [73] **PHP**: Acrónimo recursivo de PHP Hypertext Preprocessor (Preprocesador de Hipertexto), es un lenguaje de script de propósito general diseñado especialmente para desarrollar páginas web dinámicas.
- [74] **Apache2**: Servidor web de la Fundación Apache, posiblemente el más conocido y utilizado.
- [75] **Python**: Lenguaje de programación de alto nivel característico por poner especial énfasis en la claridad del código.
- [76] **Dominio de Internet**: Equivale a una dirección IP en términos memorizables y fáciles de recordar (por ejemplo *google.com* en vez de *66.249.92.104*).

- [77] **phpMyAdmin**: Herramienta de código abierto escrita en PHP para administrar bases de datos MySQL a través de la web.
- [78] **Lista de correo electrónico**: Permite la distribución masiva de correos electrónicos a quienes estén suscritos a la lista. Enviando un email a la dirección de la lista, todos los suscriptores lo recibirán de forma automática.
- [79] **Spam**: Mensajes no solicitados, normalmente enviados de forma automática y masiva, generalmente con contenido publicitario, adulto o fraudulento.
- [80] **TeamSpeak**: Software de voz sobre IP que permite a los usuarios comunicarse mediante un micrófono separados por varias salas virtuales.
- [81] **Punto de votación**: Incentivo otorgado a los jugadores cada vez que votan desde el panel de control, que les proporciona una curación gratuita dentro de una ciudad o una bendición básica de duración limitada. El máximo de puntos de votación que puede acumular un jugador es de cuatro.
- [82] **phpBB**: Abreviatura de PHP Bulletin Board (tablón de anuncios), es un foro de código libre escrito en PHP.
- [83] **Bugtracker**: Del inglés, *sistema de seguimiento de errores*, consiste en una aplicación utilizada para informar sobre los errores en un software y llevar el seguimiento de los mismos.
- [84] **Bug**: Del inglés *bicho*, se utiliza para definir un error de software. El término viene de 1947, cuando se notificó un error en un ordenador causado por una polilla.
- [85] **Falso positivo**: Se dice cuando tras realizar alguna prueba se obtiene un resultado positivo pero en realidad es negativo.
- [86] **eAthena**: Emulador del servidor de Ragnarok Online de Gravity, es el más utilizado en todos los servidores privados, seguido por Aegis.
- [87] **Aegis**: Software del servidor oficial de Ragnarok Online. Es posible que no coincida totalmente con el oficial, ya que sus versiones se van filtrando y posiblemente alguien las modifique con el tiempo, pero es mucho más fiel que eAthena.
- [88] **Fork**: Del inglés *bifurcación*, es un término para definir el hecho de copiar el código fuente de un software para comenzar un desarrollo independiente del original.

- [89] **3CeAM**: Acrónimo de Third Class eAthena Modification (*modificación de eAthena de terceras clases*), es un fork de eAthena dedicado exclusivamente a implementar las terceras clases.
- [90] **Rama**: También denominada *branch* en inglés, se trata de la copia de un desarrollo dentro de un *sistema de control de versiones*, de forma que pueda comenzarse otro desarrollo en paralelo. Esto no es un fork, ya que ambos desarrollos siguen llevándolos a cabo el mismo equipo.
- [91] **C**: Lenguaje de programación estructurada desarrollado en 1972 por Dennis Ritchie.
- [92] **DLL**: Acrónimo de Dynamic-link Library (*librería de enlace dinámico*), es un fichero con código ejecutable que se carga bajo demanda de un programa por parte del sistema operativo.
- [93] **Inyección DLL**: Técnica utilizada para ejecutar código dentro del espacio de memoria de una aplicación por medio de una DLL.
- [94] **Binario**: Sinónimo de ejecutable.
- [95] **Ragnarok Renewal**: Ampliación del Ragnarok Online indispensable para jugar.
- [96] **Diffpatcher**: Software que permite aplicar parches al ejecutable del Renewal para adaptarlo a servidores privados.
- [97] **Offset**: Del inglés *desplazamiento*, indica el número de posiciones de memoria sumadas a una dirección base para obtener una dirección absoluta específica. En general se lo puede definir como una distancia con la que obtener un punto concreto.
- [98] **NSIS**: Acrónimo de Nullsoft Scriptable Install System (*sistema de instaladores por script de Nullsoft*), software de código abierto de Nullsoft que permite crear instaladores para Windows mediante un lenguaje de script.
- [99] **Encriptación**: Técnica para alterar un mensaje de forma que no se pueda comprender su contenido hasta que se le aplique otro método que lo devuelva a su estado original.
- [100] **Paquete de datos**: Unidad básica de transporte de información dentro de una red informática.
- [101] **Parcheador**: Software encargado de aplicar parches o actualizaciones de forma automática.

- [102] **GRF**: Fichero comprimido utilizado por Gravity para almacenar mayoritariamente gráficos, audio y texto, que serán utilizados por el cliente durante el juego.
- [103] **GB**: Acrónimo de *gigabyte*, equivalente a aproximadamente mil millones de bytes.
- [104] **MB**: Acrónimo de *megabyte*, equivalente a aproximadamente un millón de bytes.
- [105] **kB**: Acrónimo de *kilobyte*, equivalente a aproximadamente mil bytes.
- [106] **CRC**: Acrónimo de *Cyclic Redundancy Check* (control de redundancia cíclica), es una función que devuelve un resultado numérico de longitud fija a partir de un flujo de datos de longitud variable. Se le llama CRC también al resultado de la función, y se suele usar para detectar alteraciones en los datos.
- [107] **Tabla (bases de datos)**: Conjunto de datos de características similares.
- [108] **C#**: Lenguaje de programación multiparadigma desarrollado por Microsoft.
- [109] **.NET Framework**: Framework de Microsoft que incluye una amplia colección de librerías para simplificar la programación de interfaces de usuario, conectividad a bases de datos, criptografía, desarrollo web, algoritmos numéricos y comunicaciones de red.
- [110] **Google AdSense**: Servicio de publicidad de Google que permite a los propietarios de páginas web mostrar anuncios de distintos tipos, con remuneraciones económicas dependiendo del número de clics e impresiones que tengan los anuncios.
- [111] **Log**: Del inglés *bitácora*, es un anglicismo para definir un registro de eventos informáticos.
- [112] **SQLite**: Librería de dominio público que implementa un motor de base de datos sin necesidad de instalar ningún servidor.
- [113] **Producto cartesiano**: Dados dos conjuntos, es la asociación de cada componente del primero con todos los componentes del segundo.
- [114] **Clave primaria**: En una base de datos relacional<sup>[183]</sup>, es el campo que define de forma única a cada fila de una tabla.
- [115] **Procedimiento almacenado**: Programa que se guarda en una base de datos y puede ser ejecutado a petición del usuario para realizar acciones directamente en el gestor de bases de datos.

- [116] **Función:** Subalgoritmo que forma parte del algoritmo principal, generalmente utilizado para resolver una tarea específica, evitar código redundante o generar un código más sencillo y mantenible.
- [117] **RAM:** Acrónimo de Random Access Memory (*memoria de acceso aleatorio*), es un sistema de almacenamiento de datos volátil utilizado por el procesador que se borra al apagar el ordenador.
- [118] **CPU:** Acrónimo de Central Processing Unit (*unidad central de proceso*), también conocido como *procesador*, es el cerebro de un ordenador, donde se realizan todas las operaciones lógicas y aritméticas y el procesamiento de datos e instrucciones de los programas.
- [119] **RAID:** Acrónimo de Redundant Array of Independent Disks (*matriz redundante de discos independientes*), se trata de una tecnología que permite almacenar datos en varios discos al mismo tiempo, ya sea como sistema de almacenamiento normal o utilizándolos como copias de seguridad replicando los datos (este último método es el que se utiliza en BidaiRO).
- [120] **Privacidad Whois:** Servicio ofrecido por algunas empresas de contratación de dominios, mediante el cual al realizar una petición whois aparecerán los datos de la empresa en vez de los del contratante.
- [121] **Swap (paginación):** En español *intercambio*, sistema para utilizar dispositivos de almacenamiento como discos duros a modo de memoria.
- [122] **Partición de disco:** División de un disco duro en múltiples discos virtuales, de forma que cada disco virtual pueda utilizarse de forma independiente.
- [123] **Partición home:** En español *casa*, partición utilizada en sistemas Linux o Unix para almacenar los datos personales de cada usuario dado de alta en el sistema. Generalmente se almacena en `/home/<usuario>`.
- [124] **Partición /:** También llamada *raíz*, es la partición principal de los sistemas Linux o Unix a partir de la cual se ramifican el resto de particiones.
- [125] **SMS:** Acrónimo de Short Message Service (*servicio de mensaje corto*), hace referencia a los mensajes de texto enviados a través de teléfonos móviles.
- [126] **Halloween:** Festividad anglosajona celebrada el 31 de octubre, originaria de la festividad celta del Samahin y la festividad cristiana del Día de todos los santos.

- [127] **Tabla temporal (base de datos):** Tabla creada de forma dinámica para almacenar información durante un periodo de tiempo limitado que se borra cuando deja de ser útil.
- [128] **Bucle:** Sentencia o conjunto de sentencias que se repiten varias veces hasta que se da una condición determinada.
- [129] **Ámbito (variables):** En programación, una variable puede ser accedida desde unas zonas u otras del código dependiendo del ámbito con que se defina. Por ejemplo, hay variables globales que pueden accederse desde todo el código, y locales que sólo pueden accederse desde la función o bloque en el que se definen. Dependiendo del lenguaje de programación puede haber más o menos ámbitos y tener distintas características.
- [130] **Método:** En programación orientada a objetos, una subrutina asociada a una clase u objeto.
- [131] **Manejador:** También llamado *handler*, en inglés, consiste en código que se ejecuta como respuesta a ciertos eventos específicos (hacer clic, cargar una página web, cerrar una ventana, etc.).
- [132] **Hilo (informática):** También llamado subproceso, es una tarea que puede ejecutarse paralelamente a otras, compartiendo recursos tales como memoria, archivos, etc.
- [133] **Conexión no interactiva (MySQL):** Tipo de conexión con el servidor que procesa una serie de comandos sin intervención por parte del usuario.
- [134] **Conexión interactiva (MySQL):** Tipo de conexión con el servidor que permite al usuario interactuar con la base de datos, además de mostrar los resultados en un formato más legible.
- [135] **Join (base de datos):** Del inglés *unión*, consiste en cruzar el contenido de dos o más tablas para poder realizar ciertos tipos de consultas.
- [136] **Índice (base de datos):** El acceso a una base de datos será mucho más rápido si se obtiene la información a través de un campo definido como índice.
- [137] **Caché:** Lugar físico donde se almacenan datos para que se pueda acceder a ellos más rápidamente.
- [138] **Buffer:** Lugar de almacenamiento de datos temporal mientras esperan a ser procesados.
- [139] **Sentencia de ordenación:** En bases de datos se trata de una sentencia para ordenar un grupo de registros devueltos por una consulta, a partir de uno o más

de sus campos. Su sintaxis es: *order by* <campo1> [<campo2>] [...] [<campoN>].

[140]**Sentencia de agrupación:** En bases de datos se trata de una sentencia para agrupar varios registros devueltos por una consulta, a partir de uno o más de sus campos. Su sintaxis es: *group by* <campo1> [<campo2>] [...] [<campoN>].



## 23 Referencias

- [1] Gravity – Global Playground Gravity [ref. de 2010]  
<http://www.gravity.co.kr/index.asp>
- [2] Ragnarokononline – Ragnarok Online [ref. de 2010]  
<http://iro.ragnarokononline.com/>
- [3] Wikipedia – Massively multiplayer online role-playing game [ref. de 2010]  
[http://en.wikipedia.org/wiki/Massively\\_multiplayer\\_online\\_role-playing\\_game](http://en.wikipedia.org/wiki/Massively_multiplayer_online_role-playing_game)
- [4] Wikipedia – Norse Mythology [ref. de 2010]  
[http://en.wikipedia.org/wiki/Norse\\_mythology#Ragnar.C3.B6k](http://en.wikipedia.org/wiki/Norse_mythology#Ragnar.C3.B6k)
- [5] Wikipedia – Ragnarok Online [ref. de 2010]  
[http://en.wikipedia.org/wiki/Ragnarok\\_Online](http://en.wikipedia.org/wiki/Ragnarok_Online)
- [6] Wikipedia – Open source [ref. de 2010]  
[http://es.wikipedia.org/wiki/Open\\_source](http://es.wikipedia.org/wiki/Open_source)
- [7] Subversion – subversión.tigris.org [ref. de 2009] <http://subversion.tigris.org/>
- [8] Wikipedia – Revision control [ref. de 2010]  
[http://en.wikipedia.org/wiki/Revision\\_control](http://en.wikipedia.org/wiki/Revision_control)
- [9] Wikipedia – Tubo de rayos catódicos [ref. de 2010]  
[http://es.wikipedia.org/wiki/Tubo\\_de\\_rayos\\_cat%C3%B3dicos](http://es.wikipedia.org/wiki/Tubo_de_rayos_cat%C3%B3dicos)
- [10] Wikipedia – OXO [ref. de 2010] <http://en.wikipedia.org/wiki/OXO>
- [11] Wikipedia – Spacewar! [ref. de 2010] <http://en.wikipedia.org/wiki/Spacewar!>
- [12] Wikipedia – PDP-1 [ref. de 2010] <http://en.wikipedia.org/wiki/PDP-1>
- [13] Wikipedia – Light gun [ref. de 2010] [http://en.wikipedia.org/wiki/Light\\_gun](http://en.wikipedia.org/wiki/Light_gun)
- [14] Nintendo – Inicio [ref. de 2010] <http://www.nintendo.es/>
- [15] Wikipedia – Nintendo [ref. de 2010] <http://en.wikipedia.org/wiki/Nintendo>
- [16] Wikipedia – Duck Hunt [ref. de 2010] [http://en.wikipedia.org/wiki/Duck\\_Hunt](http://en.wikipedia.org/wiki/Duck_Hunt)
- [17] Wikipedia – Videoconsola [ref. de 2010]  
<http://es.wikipedia.org/wiki/Videoconsola>
- [18] Wikipedia – Galaxy Game [ref. de 2010]  
[http://en.wikipedia.org/wiki/Galaxy\\_Game](http://en.wikipedia.org/wiki/Galaxy_Game)
- [19] Wikipedia – Atari, Inc. [ref. de 2010] [http://en.wikipedia.org/wiki/Atari,\\_Inc](http://en.wikipedia.org/wiki/Atari,_Inc)
- [20] Wikipedia – Pong [ref. de 2010] <http://en.wikipedia.org/wiki/Pong>

- [21] Wikipedia – Space Invaders [ref. de 2010]  
[http://en.wikipedia.org/wiki/Space\\_invaders](http://en.wikipedia.org/wiki/Space_invaders)
- [22] Wikipedia – Asteroids (video game) [ref. de 2010]  
[http://en.wikipedia.org/wiki/Asteroids\\_%28video\\_game%29](http://en.wikipedia.org/wiki/Asteroids_%28video_game%29)
- [23] Wikipedia – Pac-Man [ref. de 2010] <http://en.wikipedia.org/wiki/Pacman>
- [24] Wikipedia – Magnavox Odyssey [ref. de 2010]  
[http://en.wikipedia.org/wiki/Magnavox\\_Odyssey](http://en.wikipedia.org/wiki/Magnavox_Odyssey)
- [25] Wikipedia – Magnavox [ref. de 2010] <http://en.wikipedia.org/wiki/Magnavox>
- [26] Wikipedia – Atari 2600 [ref. de 2010] [http://en.wikipedia.org/wiki/Atari\\_2600](http://en.wikipedia.org/wiki/Atari_2600)
- [27] Wikipedia – Intellivision [ref. de 2010]  
<http://en.wikipedia.org/wiki/Intellivision>
- [28] Wikipedia – Mattel [ref. de 2010] <http://en.wikipedia.org/wiki/Mattel>
- [29] Mattel – Mattel Inc. [ref. de 2010] <http://www.mattel.com/>
- [30] Wikipedia – ColecoVision [ref. de 2010]  
<http://en.wikipedia.org/wiki/ColecoVision>
- [31] Wikipedia – Coleco [ref. de 2010] <http://en.wikipedia.org/wiki/Coleco>
- [32] Coleco Museum – Dedicated to COLECO collectibles. [ref. de 2010]  
<http://www.colecomuseum.com/>
- [33] Wikipedia – Commodore 64 [ref. de 2010]  
[http://en.wikipedia.org/wiki/Commodore\\_64](http://en.wikipedia.org/wiki/Commodore_64)
- [34] C64.COM – To Protect and Preserve [ref. de 2010] <http://www.c64.com/>
- [35] Wikipedia – Commodore International [ref. de 2010]  
[http://en.wikipedia.org/wiki/Commodore\\_International](http://en.wikipedia.org/wiki/Commodore_International)
- [36] Commodoregaming – Commodore Gaming webshop [ref. de 2007]  
<http://www.commodoregaming.com/>
- [37] Wikipedia – Apple II series [ref. de 2010]  
[http://en.wikipedia.org/wiki/Apple\\_II\\_series](http://en.wikipedia.org/wiki/Apple_II_series)
- [38] Wikipedia – Apple Inc. [ref. de 2010] [http://en.wikipedia.org/wiki/Apple\\_Inc.](http://en.wikipedia.org/wiki/Apple_Inc.)
- [39] Apple – Apple [ref. de 2010] <http://www.apple.com/>
- [40] Wikipedia – ZX Spectrum [ref. de 2010]  
[http://en.wikipedia.org/wiki/ZX\\_Spectrum](http://en.wikipedia.org/wiki/ZX_Spectrum)
- [41] Wikipedia – Sinclair Research [ref. de 2010]  
[http://en.wikipedia.org/wiki/Sinclair\\_Research\\_Ltd](http://en.wikipedia.org/wiki/Sinclair_Research_Ltd)

- [42] Sinclair-research – Sinclair Research [ref. de 2009] <http://www.sinclair-research.co.uk/>
- [43] Wikipedia – Amstrad CPC [ref. de 2010] [http://en.wikipedia.org/wiki/Amstrad\\_CPC](http://en.wikipedia.org/wiki/Amstrad_CPC)
- [44] Wikipedia – Amstrad [ref. de 2010] <http://en.wikipedia.org/wiki/Amstrad>
- [45] Amstrad – Welcome to Amstrad Plc [ref. de 2008] <http://www.amstrad.com/>
- [46] Amstrad-esp – Amstrad ESP [ref. de 2010] <http://www.amstrad-esp.com/>
- [47] Wikipedia – Atari ST [ref. de 2010] [http://en.wikipedia.org/wiki/Atari\\_ST](http://en.wikipedia.org/wiki/Atari_ST)
- [48] Wikipedia – Amiga [ref. de 2010] <http://en.wikipedia.org/wiki/Amiga>
- [49] Wikipedia – Nintendo Entertainment System [ref. de 2010] [http://en.wikipedia.org/wiki/Nintendo\\_Entertainment\\_System](http://en.wikipedia.org/wiki/Nintendo_Entertainment_System)
- [50] Wikipedia – Sega Master System [ref. de 2010] [http://en.wikipedia.org/wiki/Sega\\_master\\_system](http://en.wikipedia.org/wiki/Sega_master_system)
- [51] Sega – SEGA [ref. de 2010] <http://www.sega.es/>
- [52] Wikipedia – Alone in the Dark [ref. de 2010] [http://en.wikipedia.org/wiki/Alone\\_in\\_the\\_dark](http://en.wikipedia.org/wiki/Alone_in_the_dark)
- [53] Wikipedia – King’s Quest [ref. de 2010] [http://en.wikipedia.org/wiki/Kings\\_Quest](http://en.wikipedia.org/wiki/Kings_Quest)
- [54] Wikipedia – The Secret of Monkey Island [ref. de 2010] [http://en.wikipedia.org/wiki/The\\_Secret\\_of\\_Monkey\\_Island](http://en.wikipedia.org/wiki/The_Secret_of_Monkey_Island)
- [55] Wikipedia – Myst [ref. de 2010] <http://en.wikipedia.org/wiki/Myst>
- [56] Wikipedia – Sim City [ref. de 2010] [http://en.wikipedia.org/wiki/Sim\\_city](http://en.wikipedia.org/wiki/Sim_city)
- [57] Wikipedia – Maxis [ref. de 2010] <http://en.wikipedia.org/wiki/Maxis>
- [58] Wikipedia – Heroes of Might and Magic: A Strategic Quest [ref. de 2010] [http://en.wikipedia.org/wiki/Heroes\\_of\\_Might\\_and\\_Magic: A Strategic Quest](http://en.wikipedia.org/wiki/Heroes_of_Might_and_Magic:_A_Strategic_Quest)
- [59] Wikipedia – Game Boy [ref. de 2010] [http://en.wikipedia.org/wiki/Game\\_boy](http://en.wikipedia.org/wiki/Game_boy)
- [60] Wikipedia – Sega Game Gear [ref. de 2010] [http://en.wikipedia.org/wiki/Game\\_gear](http://en.wikipedia.org/wiki/Game_gear)
- [61] Wikipedia – Atari Lynx [ref. de 2010] [http://en.wikipedia.org/wiki/Atari\\_Lynx](http://en.wikipedia.org/wiki/Atari_Lynx)
- [62] Wikipedia – Nokia [ref. de 2010] <http://en.wikipedia.org/wiki/Nokia>
- [63] Nokia – Nokia on the Web [ref. de 2010] <http://www.nokia.com/>
- [64] Wikipedia – Snake (video game) [ref. de 2010] [http://en.wikipedia.org/wiki/Snake %28video game%29](http://en.wikipedia.org/wiki/Snake_%28video_game%29)

- [65] Wikipedia – Mega Drive [ref. de 2010]  
[http://en.wikipedia.org/wiki/Mega\\_Drive](http://en.wikipedia.org/wiki/Mega_Drive)
- [66] Wikipedia – Super Nintendo Entertainment System [ref. de 2010]  
[http://en.wikipedia.org/wiki/Super\\_nintendo](http://en.wikipedia.org/wiki/Super_nintendo)
- [67] Wikipedia – TurboGrafx-16 [ref. de 2010]  
<http://en.wikipedia.org/wiki/Turbografx-16>
- [68] Wikipedia – NEC [ref. de 2010] <http://en.wikipedia.org/wiki/Nec>
- [69] NEC – NEC Global [ref. de 2010] <http://www.nec.com/>
- [70] Wikipedia – Neo Geo (console) [ref. de 2010]  
[http://en.wikipedia.org/wiki/Neo\\_Geo\\_%28console%29](http://en.wikipedia.org/wiki/Neo_Geo_%28console%29)
- [71] Wikipedia – SNK Playmore [ref. de 2010] <http://en.wikipedia.org/wiki/Snk>
- [72] Snkplaymoreusa – SNK PLAYMORE USA [ref. de 2010]  
<http://www.snkplaymoreusa.com/>
- [73] Wikipedia – Sega Saturn [ref. de 2010]  
[http://en.wikipedia.org/wiki/Sega\\_saturn](http://en.wikipedia.org/wiki/Sega_saturn)
- [74] Wikipedia – Nintendo 64 [ref. de 2010]  
[http://en.wikipedia.org/wiki/Nintendo\\_64](http://en.wikipedia.org/wiki/Nintendo_64)
- [75] Wikipedia – PlayStation [ref. de 2010]  
<http://en.wikipedia.org/wiki/PlayStation>
- [76] Wikipedia – Sony [ref. de 2010] <http://en.wikipedia.org/wiki/Sony>
- [77] Sony – Sony Global Headquarters [ref. de 2010] <http://www.sony.net/>
- [78] Wikipedia – Half-Life (video game) [ref. de 2010]  
[http://en.wikipedia.org/wiki/Half-Life\\_%28video\\_game%29](http://en.wikipedia.org/wiki/Half-Life_%28video_game%29)
- [79] Wikipedia – Valve Corporation [ref. de 2010]  
[http://en.wikipedia.org/wiki/Valve\\_Corporation](http://en.wikipedia.org/wiki/Valve_Corporation)
- [80] Valvesoftware – Valve [ref. de 2010] <http://www.valvesoftware.com/>
- [81] Wikipedia – Counter-Strike [ref. de 2010]  
<http://en.wikipedia.org/wiki/Counter-Strike>
- [82] Wikipedia – Baldur's Gate II: Shadows of Amn [ref. de 2010]  
[http://en.wikipedia.org/wiki/Baldur%27s\\_Gate\\_II:\\_Shadows\\_of\\_Amn](http://en.wikipedia.org/wiki/Baldur%27s_Gate_II:_Shadows_of_Amn)
- [83] Maps4heroes – Heroes 5. Heroes 4. Heroes 3. Heroes maps [ref. de 2010]  
<http://www.maps4heroes.com/>
- [84] Celestialheavens – All Things Might and Magic [ref. de 2010]  
<http://www.celestialheavens.com/>

- [85] Wikipedia – Second Life [ref. de 2010]  
[http://en.wikipedia.org/wiki/Second\\_Life](http://en.wikipedia.org/wiki/Second_Life)
- [86] Secondlife – Second Life Official Site [ref. de 2010] <http://secondlife.com/>
- [87] Wikipedia – Xbox [ref. de 2010] <http://en.wikipedia.org/wiki/Xbox>
- [88] Wikipedia – Microsoft [ref. de 2010] <http://en.wikipedia.org/wiki/Microsoft>
- [89] Microsoft – Microsoft Worlwide Homre [ref. de 2010]  
<http://www.microsoft.com/worldwide/>
- [90] Wikipedia – PlayStation2 [ref. de 2010]  
<http://en.wikipedia.org/wiki/Playstation2>
- [91] Wikipedia – Nintendo GameCube [ref. de 2010]  
<http://en.wikipedia.org/wiki/Gamecube>
- [92] Wikipedia – Xbox 360 [ref. de 2010] [http://en.wikipedia.org/wiki/Xbox\\_360](http://en.wikipedia.org/wiki/Xbox_360)
- [93] Wikipedia – PlayStation3 [ref. de 2010]  
<http://en.wikipedia.org/wiki/Playstation3>
- [94] Wikipedia – Nintendo DS [ref. de 2010]  
[http://en.wikipedia.org/wiki/Nintendo\\_ds](http://en.wikipedia.org/wiki/Nintendo_ds)
- [95] Wikipedia –Wii [ref. de 2010] <http://en.wikipedia.org/wiki/Wii>
- [96] Wikipedia – Play-by-mail game [ref. de 2010]  
[http://en.wikipedia.org/wiki/Play-by-mail\\_game](http://en.wikipedia.org/wiki/Play-by-mail_game)
- [97] Wikipedia – Reglamento del ajedrez [ref. de 2010]  
[http://es.wikipedia.org/wiki/Reglamento\\_del\\_ajedrez](http://es.wikipedia.org/wiki/Reglamento_del_ajedrez)
- [98] Ageofwonders – Age of Wonders [ref. de 2010]  
<http://www.ageofwonders.com/>
- [99] Vgaplanets – VGA Planets Home Page [ref. de 2005]  
<http://www.vgaplanets.com/>
- [100] Wikipedia – Play-by-Internet [ref. de 2010] <http://en.wikipedia.org/wiki/Play-by-Internet>
- [101] Google – Google Directory [ref. de 2010]  
<http://www.google.com/Top/Games/Online/Play-By-E-Mail/>
- [102] Wikipedia – MUD [ref. de 2010] <http://en.wikipedia.org/wiki/MUD>
- [103] Wikipedia – Adventure computer game [ref. de 2010]  
[http://en.wikipedia.org/wiki/Adventure\\_%28computer\\_game%29](http://en.wikipedia.org/wiki/Adventure_%28computer_game%29)
- [104] Wikipedia – La Aventura Original [ref. de 2010]  
[http://es.wikipedia.org/wiki/La\\_Aventura\\_Original](http://es.wikipedia.org/wiki/La_Aventura_Original)

- [105] Wikipedia – Dungeons & Dragons [ref. de 2010]  
[http://en.wikipedia.org/wiki/Dungeons\\_%26\\_Dragons](http://en.wikipedia.org/wiki/Dungeons_%26_Dragons)
- [106] Wizards – Dungeons & Dragons Roleplaying Game Official Home Page [ref. de 2010] <http://www.wizards.com/dnd/>
- [107] Wikipedia – Zork [ref. de 2010] <http://en.wikipedia.org/wiki/Zork>
- [108] Wikipedia – MUD1 [ref. de 2010] <http://en.wikipedia.org/wiki/MUD1>
- [109] Wikipedia – AberMUD [ref. de 2010] <http://en.wikipedia.org/wiki/AberMUD>
- [110] Wikipedia – B programming language [ref. de 2010]  
[http://en.wikipedia.org/wiki/B\\_%28programming\\_language%29](http://en.wikipedia.org/wiki/B_%28programming_language%29)
- [111] Wikipedia – C programming language [ref. de 2010]  
[http://en.wikipedia.org/wiki/C\\_%28programming\\_language%29](http://en.wikipedia.org/wiki/C_%28programming_language%29)
- [112] Wikipedia – Unix [ref. de 2010] <http://en.wikipedia.org/wiki/Unix>
- [113] Wikipedia – TinyMUD [ref. de 2010] <http://en.wikipedia.org/wiki/TinyMUD>
- [114] Wikipedia – LPMud [ref. de 2010] <http://en.wikipedia.org/wiki/LPMud>
- [115] Wikipedia – DikuMUD [ref. de 2010] <http://en.wikipedia.org/wiki/DikuMUD>
- [116] Zuggsoft – Zugg Software :: CMUD [ref. de 2009]  
<http://forums.zuggsoft.com/index.php?p=cmud>
- [117] Zuggsoft – Zugg Software :: Home [ref. de 2009]  
<http://www.zuggsoft.com/index.php>
- [118] Rlmud – Reinos de Leyenda. El MUD en español gratuito [ref. de 2010]  
<http://www.rlmud.org/>
- [119] Mudconnect – The MUD Connector [ref. de 2009]  
<http://www.mudconnect.com/>
- [120] Wikipedia – Spasim [ref. de 2010] <http://en.wikipedia.org/wiki/Spasim>
- [121] Wikipedia – Maze War [ref. de 2010] [http://en.wikipedia.org/wiki/Maze\\_War](http://en.wikipedia.org/wiki/Maze_War)
- [122] Wikipedia – First person video games [ref. de 2010]  
[http://en.wikipedia.org/wiki/First\\_person\\_%28video\\_games%29](http://en.wikipedia.org/wiki/First_person_%28video_games%29)
- [123] Wikipedia – PLATO [ref. de 2010]  
[http://en.wikipedia.org/wiki/PLATO\\_network](http://en.wikipedia.org/wiki/PLATO_network)
- [124] Wikipedia – Star Wars: X-Wing computer game series [ref. de 2010]  
[http://en.wikipedia.org/wiki/Star\\_Wars:\\_X-Wing\\_computer\\_game\\_series](http://en.wikipedia.org/wiki/Star_Wars:_X-Wing_computer_game_series)
- [125] Wikipedia – Marathon Trilogy [ref. de 2010]  
[http://en.wikipedia.org/wiki/Marathon\\_Triology](http://en.wikipedia.org/wiki/Marathon_Triology)
- [126] Wikipedia – Unreal [ref. de 2010] <http://en.wikipedia.org/wiki/Unreal>

- [127] Worldofwarcraft – World of Warcraft [ref. de 2010]  
<http://www.worldofwarcraft.com/index.xml>
- [128] Wikipedia – Utopia video game [ref. de 2010]  
[http://en.wikipedia.org/wiki/Utopia\\_%28video\\_game%29](http://en.wikipedia.org/wiki/Utopia_%28video_game%29)
- [129] Wikipedia – Herzog Zwei [ref. de 2010]  
[http://en.wikipedia.org/wiki/Herzog\\_Zwei](http://en.wikipedia.org/wiki/Herzog_Zwei)
- [130] Blizzard – Blizzard Entertainment: Starcraft [ref. de 2010]  
<http://eu.blizzard.com/es-es/games/sc/index.html?rhtml=y>
- [131] Blizzard – Blizzard Entertainment [ref. de 2010] <http://eu.blizzard.com/es-es/>
- [132] Battle – Battle.net [ref. de 2010] <http://eu.battle.net/?rhtml=y>
- [133] Mightandmagic – Heroes of Might and Magic V [ref. de 2006]  
<http://www.mightandmagic.com/HeroesV/us/home.php>
- [134] Ubi – Ubisoft [ref. de 2008] <http://www.ubi.com>
- [135] Java – java.com [ref. de 2010] <http://www.java.com/es/>
- [136] Ogame – Ogame Homepage [ref. de 2010] <http://www.ogame.org/>
- [137] Travian – Travian [ref. de 2010] <http://www.travian.net/>
- [138] Wikipedia – Tetris [ref. de 2010] <http://es.wikipedia.org/wiki/Tetris>
- [139] Tetrisfriends – Play Free Tetris games online at Tetris Friends [ref. de 2010]  
<http://www.tetrisfriends.com/>
- [140] Forsvarsmakten – The Sweddish Armed Forces [ref. de 2010]  
<http://team.forsvarsmakten.se/english/>
- [141] Wikipedia – Ultima Online [ref. de 2010]  
[http://en.wikipedia.org/wiki/Ultima\\_Online](http://en.wikipedia.org/wiki/Ultima_Online)
- [142] Wikipedia – Neverwinter Nights (AOL game) [ref. de 2010]  
[http://en.wikipedia.org/wiki/Neverwinter\\_Nights\\_%28AOL\\_game%29](http://en.wikipedia.org/wiki/Neverwinter_Nights_%28AOL_game%29)
- [143] Wikipedia – Meridian 59 [ref. de 2010]  
[http://en.wikipedia.org/wiki/Meridian\\_59](http://en.wikipedia.org/wiki/Meridian_59)
- [144] Guildwars – Welcome to the Official Guild Wars Website [ref. de 2010]  
<http://www.guildwars.com/>
- [145] Rune-nifelheim – Quests generals y lista [ref. de 2010] <http://rune-nifelheim.com/quests.php>
- [146] Foros de Irowiki – Ragnarok Renewal Discussion Thread II [ref. de 2008].  
 Disponible en <http://forums.irowiki.org/showthread.php?t=15359>

- [147] Wikipedia – Ragnarok Online 2: The Gate of the World [ref. de 2010]  
[http://en.wikipedia.org/wiki/Ragnarok\\_Online\\_2:\\_The\\_Gate\\_of\\_the\\_World](http://en.wikipedia.org/wiki/Ragnarok_Online_2:_The_Gate_of_the_World)
- [148] Wikipedia – Ragnarok Online 2: Legend of the Second [ref. de 2010]  
[http://en.wikipedia.org/wiki/Ragnarok\\_Online\\_2](http://en.wikipedia.org/wiki/Ragnarok_Online_2)
- [149] Openkore Wiki – Official server types [ref. de 2010]  
<http://wiki.openkore.com/index.php?title=ServerType>
- [150] Ragnarok Online (iRO) – Premium benefits [ref. de 2006]  
<http://iro.ragnarokonline.com/payment/subscription.asp>
- [151] Ragnarok Online (iRO) – FAQ [ref. de 2006]  
<http://iro.ragnarokonline.com/support/questioncontent.asp?noticeSeqNo=156&Page=9&Condition=&KeyWord=&mNum=0>
- [152] Ragnarok Online (euRO) – FAQ [ref. de 2004] <https://www.euro-ro.net/support/faq/eu/00033/>
- [153] Rofitorium – Top Servers RO [ref. de 2010] <http://www.rofitorium.com/top/>
- [154] Devstorm Network – Top Ragnarok Online [ref. de 2010] <http://top.devstorm-network.org/>
- [155] DivineRO – Nezumi Top Server [ref. de 2010]  
<http://www.nezumi.divinero.net/>
- [156] Rune Nifelheim – Top Ragnarok Online [ref. de 2010]  
<http://top.ragnarokonline.es/>
- [157] FreeLists – Listas de correo gratuitas [ref. de 2010] <http://www.freelists.org/>
- [158] DomainIt – Register or Backorder a Domain [ref. de 2010]  
<http://www.domainit.com/>
- [159] Google – Google [ref. de 2010] <http://www.google.com/>
- [160] Gmail – Google Mail [ref. de 2010] <http://www.gmail.com>
- [161] TeamSpeak – TeamSpeak Communication System [ref. de 2010]  
<http://www.teamspeak.com>
- [162] Sourceforge – Ceres CP [ref. de 2008] <http://cerescp.sourceforge.net/>
- [163] phpBB – Open Source Bulletin Board Software [ref. de 2007]  
<http://www.phpbb.com/>
- [164] eAthena – eAthena Support Board [ref. de 2010]  
<http://www.eathena.ws/board/>
- [165] Aegis – Aegis Support Board [ref. de 2010] <http://forum.asb-sakray.net/>



- [166] 3CeAM – Third Class eAthena Modification [ref. de 2010]  
<http://www.3ceam.com/>
- [167] Wikipedia – DLL Injection [ref. de 2010]  
[http://en.wikipedia.org/wiki/DLL\\_injection](http://en.wikipedia.org/wiki/DLL_injection)
- [168] Foro eAthena – kRO Diffs by Diffteam™ [ref. de 2010]  
<http://www.eathena.ws/board/index.php?showtopic=249268>
- [169] Nullsoft scriptable install system – Main Page [ref. de 2010]  
[http://nsis.sourceforge.net/Main\\_Page](http://nsis.sourceforge.net/Main_Page)
- [170] Creative Commons – Creative Commons [ref. de 2010]  
<http://creativecommons.org/>
- [171] Foro eAthena – Ragnarok Renewal English Data [ref. de 2010]  
<http://www.eathena.ws/board/index.php?showtopic=230035&hl=renewal>
- [172] Sourceforge – GRF Tool [ref. de 2005]  
<http://sourceforge.net/projects/openkore/files/GRF%20Tool/1.2.0/>
- [173] Megaupload – Gryff [ref. de 2005]  
<http://www.megaupload.com/?d=FA93RUKN>
- [174] Modpython – The Apache Software Foundation [ref. de 2008]  
<http://www.modpython.org/>
- [175] Google – Google AdSense [ref. de 2010] <https://www.google.com/adsense/>
- [176] MySQL – MySQL :: The world's most popular open source database [ref. de 2010] <http://mysql.com/>
- [177] PostgreSQL – PostgreSQL :: The world's most advanced open source database [ref. de 2010] <http://www.postgresql.org/>
- [178] MySQL – MySQL 5.0 Reference Manual :: 7 Optimization [ref. de 2010]  
<http://dev.mysql.com/doc/refman/5.0/en/optimization.html>
- [179] Genomewiki – Tuning-premier.sh [ref. de 2010]  
<http://genomewiki.ucsc.edu/index.php/Tuning-primer.sh>
- [180] MySQLTuner – High-preformance MySQL optimization script [ref. de 2010]  
<http://blog.mysqltuner.com/>
- [181] SQLite – SQLite Home Page [ref. de 2010] <http://www.sqlite.org/>
- [182] RateMyServer.Net – RO Database & Server Reviews [ref. de 2010]  
<http://ratemyserver.net/>
- [183] Wikipedia – Relational database [ref. de 2010]  
[http://en.wikipedia.org/wiki/Relational\\_database](http://en.wikipedia.org/wiki/Relational_database)